

Program Fortran Untuk Menghasilkan Gambar Grafik Dalam Fisika Komputasional

Arief Hermanto

Jurusan Fisika, FMIPA-UGM

Abstrak

Dalam makalah ini kami melaporkan penyusunan program dalam Fortran 77 untuk menghasilkan grafik, baik untuk kerja analisis maupun untuk tampilan yang persuasif. Dengan tambahan perlengkapan ini maka Fortran 77, meskipun sudah cukup tua, masih tetap merupakan peralatan yang handal dalam fisika komputasional.

Kata kunci : program, grafik, Fortran 77

Abstract

Fortran code to produce graphical picture in Computational Physics. In this article we report the construction of Fortran codes to produce graphical picture for analytical work or persuasive presentation as well. With this additional software we conclude that Fortran 77, although rather old, can still be a reliable main tool in computational physics.

Keywords : code, graphics, Fortran 77

1. Pendahuluan

Fisika merupakan salah satu pilar penyangga peradaban manusia. Komputer dilahirkan oleh fisika. Pada mulanya komputer merupakan alat hitung mekanik. Penemuan transistor yang berbasis semikonduktor menjadi awalan yang dengan pengembangan yang kontinu memungkinkan komputer bekerja dengan cepat sehingga dapat berfungsi optimal seperti sekarang ini. Saat ini mulai dikembangkan komputer kuantum dimana tahapan teoretik sudah cukup lanjut sedangkan tahapan eksperimental boleh dikatakan masih pada tahap awal tapi banyak memberikan harapan ke depan.

Karena fisika membidani lahirnya komputer maka sudah sewajarnya komputer kemudian sangat dimanfaatkan dalam pengembangan fisika. Mula-mula komputer digunakan sebagai alat bantu hitung (misalnya untuk menentukan solusi persamaan nonlinear, menentukan nilai integral, menentukan solusi persamaan diferensial) dan kemudian digunakan dalam simulasi besar-besaran sehingga ada yang menempatkan fisika komputasional pada posisi sejajar dengan fisika teoretik dan fisika eksperimental.

Dalam pemanfaatan komputer dalam fisika kita memberikan perintah kepada komputer untuk melakukan apa yang kita inginkan dengan menggunakan bahasa yang dapat dimengerti oleh kita dan komputer. Seperti halnya bahasa manusia keinginan berkomunikasi itu kemudian memunculkan aneka warna bahasa pemrograman. Dengan adanya banyak alternatif itu maka kita harus menentukan pilihan bahasa pemrograman apa yang akan kita gunakan dalam tugas mulia kita sebagai anggota masyarakat fisika, yaitu menjaga dan mengembangkan fisika sebagai salah satu pilar masyarakat.

2. Bahasa pemrograman Fortran 77

2.1. Pentingnya bahasa

Bahwa bahasa sangat penting bagi manusia tidaklah perlu dibuktikan lagi (merupakan suatu postulat). Dalam komunikasi antar manusia ada banyak sekali bahasa verbal. Bahasa merupakan sarana berpikir manusia. Banyak ilmu yang berlandaskan bahasa verbal (sehingga mungkin perlu dibedakan antara bahasa verbal sehari-hari dengan bahasa verbal teknis keilmuan). Matematika juga merupakan sebuah bahasa (Devlin, 2000). Fisika merupakan ilmu yang berlandaskan matematika. Tentu saja sebagai bahasa, matematika juga merupakan sarana berpikir yang sangat handal. Galileo menyatakan bahwa rahasia keteraturan alam semesta dituliskan dalam bahasa matematika. Jadi, matematika adalah bahasa komunikasi antara manusia dengan alam semesta. Seandainya alam semesta bisa berbicara, maka dia hanya akan menjawab pertanyaan kita kalau kita berbicara padanya dengan bahasa matematika.

Bahasa pemrograman adalah alat komunikasi antara manusia dengan komputer. Karena sudah mencapai tingkat kecanggihan yang tinggi maka komputer tidak lagi hanya sekedar alat bantu hitung tapi sudah menjadi sesuatu yang bisa dianggap mempunyai posisi persis di bawah manusia (bahkan mungkin hampir sejajar) dalam tatanan tingkatan mahluk di alam semesta. Salah satu hal yang menguatkan kesan itu adalah adanya bahasa pemrograman yang beraneka macam. Kita kadang berkomunikasi dengan hewan tapi tidak menggunakan bahasa; kita hanya menggunakan isyarat. Komunikasi kita dengan komputer mungkin memang hanya isyarat juga, tapi karena sudah begitu canggih maka bisa dikatakan sudah menjadi bahasa (yang kemudian disebut sebagai bahasa pemrograman).

Sudah cukup kuat rasanya alasan untuk dapat menerima bahasa pemrograman pada posisi hampir sama dengan bahasa manusia. Perbedaan yang sangat jelas adalah tidak adanya keraguan dalam bahasa pemrograman. Bahasa manusia memang harus mengandung keraguan sehingga memungkinkan disusunnya bahasa puisi yang kaya akan nuansa dan mampu untuk mendentingkan senar-senar keindahan dalam jiwa manusia. Kita dapat berkikah balik bahwa bahasa pemrograman juga dapat memenuhi keinginan manusia akan keindahan puisi :

Why is programming fun ? What delights may its practitioner expect as his reward ? ...there is the delight of working in such a tractable medium. The programmer, like the poet, works only slightly removed from pure thought-stuff... Yet the program construct, unlike the poet's words, is real in the sense that it moves and works, producing visible outputs separate from the construct itself... (Miller, 2004)

Matematika berada di tengah antara bahasa verbal dan bahasa pemrograman. Bahasa matematika tidaklah kaku sekali seperti bahasa pemrograman namun jauh lebih eksak dari pada bahasa verbal untuk menghilangkan ambiguitas.

Jika kita dapat menerima kesejajaran antara bahasa pemrograman dengan bahasa manusia maka kita berhadapan dengan kenyataan bahwa dalam himpunan bahasa manusia, bahasa ibu mempunyai posisi penting. Seseorang dapat menguasai beberapa bahasa manusia dengan sangat lancar namun tetap saja bahasa ibunya (yaitu bahasa yang pertama kali dipelajarinya ketika balita) mempunyai tempat penting dalam jiwanya. Kita dapat menguasai beberapa bahasa pemrograman dengan baik, namun bahasa pemrograman pertama yang kita kuasai ketika kita berubah fase dari buta-pemrograman menjadi melek-pemrograman tetap menempati posisi penting dalam diri kita.

2.2. Fortran 77

Saat ini dikenal banyak sekali bahasa pemrograman, misalnya : Basic, Pascal, Delphi, Java, C, C++, Matlab, Scilab, Freemat, Octave, Maple, Maxima, Macsyma, Lisp, Haskell, Prolog dan lain-lainnya. Di antara himpunan bahasa pemrograman itu Fortran merupakan bahasa yang tua karena lahir pada awal munculnya komputer dan terutama digunakan dalam komputasi numerik dalam sains dan teknik. Secara garis besar ada 2 generasi Fortran yaitu Fortran 77 dan Fortran 90. Fortran 90 (yang bahkan lebih diperbarui menjadi Fortran 2003) dimaksudkan untuk menghadapi tantangan berbagai bahasa lain yang lahir lebih kini.

Seperti halnya bahasa manusia, pada dasarnya antara berbagai bahasa itu kita tidak dapat menentukan manakah bahasa yang terbaik. Ada kalanya orang menilai suatu bahasa berdasarkan materi yang dibicarakan. Misalnya banyak yang mengatakan bahwa bahasa Indonesia dinilai lemah ketika kita berbicara tentang sains dan teknologi. Namun bahasa Indonesia adalah bahasa ibu kita. Itu merupakan alasan yang cukup kuat untuk tetap bertahan menggunakan bahasa Indonesia ketika menulis makalah ini. Kelemahan suatu bahasa dalam satu segi selalu dapat diadaptasi oleh pemakainya dengan menggunakan berbagai inovasi.

Dalam komunikasi dengan komputer kita harus mempunyai pilihan bahasa apa yang kita pakai. Salah satu alasan kuat adalah jika bahasa pilihan itu merupakan bahasa pemrograman ibu bagi kita. Seandainya kita berada dalam daerah komputasi numerik dan bahasa ibu kita adalah Fortran 77, maka pilihan itu cukup beralasan. Fortran 77 memang mempunyai kekuatan dalam komputasi numerik. Meskipun tua namun masyarakat pemakainya masih cukup besar. Lagi pula Fortran 77 merupakan *subset* dari Fortran 90/95 sehingga semua program dalam Fortran 77 masih tetap dijalankan dengan Fortran 90/95. Masih ada kuliah yang mengajarkan bahasa Fortran 77 (Langtangen, 2004) dan masih ada buku baru yang menggunakan Fortran 77 (Date, 2005). Beberapa keunggulan bahasa Fortran 77 di antaranya adalah : sederhana, mudah dipelajari, efisien

dalam komputasi numerik, *library* yang luas, *compiler* yang optimal dalam penanganan *array*. Berbagai kelemahan dari Fortran 77 seperti tipe variabel yang terbatas dan tidak adanya alokasi memori yang dinamik sudah diatasi dalam Fortran 90/95.

Sementara itu meskipun Fortran 90/95 dimaksudkan untuk menjawab tantangan jaman dengan memasukkan fitur-fitur modern namun efisiensinya belum teruji dengan meyakinkan. Pengujian efisiensi atau kecepatan masih menunjukkan bahwa Fortran 77 tidak kalah cepat dibandingkan dengan yang lain (Decyk, 1995). Bahasa-bahasa "kelas berat" seperti Matlab dan Maple (yang memerlukan tempat sekitar 1 GB atau lebih) masih memperhatikan keberadaan Fortran 77 dengan adanya beberapa fasilitas yang berkaitan. Misalnya Maple mempunyai fasilitas untuk mengubah sebuah bentuk persamaan ke dalam Fortran, sebagai berikut.

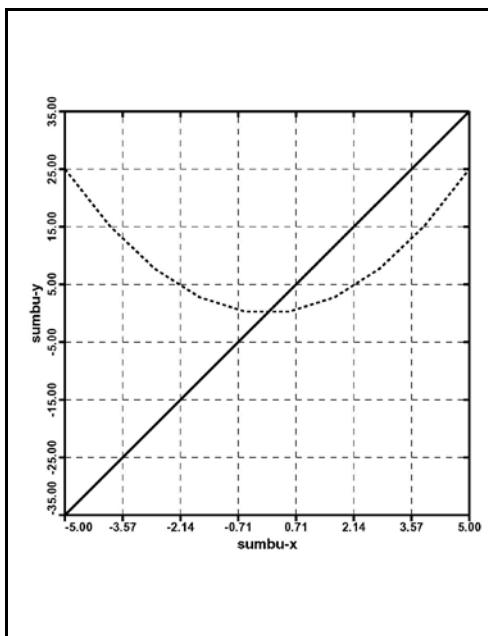
```
> f := 1-2*x+3*x^2-2*x^3+x^4;  
  
> fortran(f,optimized);  
  
t2 = x**2  
t6 = t2**2  
t7 = 1-2*x+3*t2-2*t2*x+t6  
  
> g := x->1-2*x+3*x^2-2*x^3+x^4;  
  
> fortran(g);  
  
c The options were : operatorarrow  
real function g(x)  
real x  
g = 1-2*x+3*x**2-2*x**3+x**4  
return  
end
```

3. Grafik dalam Fortran 77

3.1. Grafik untuk analisis dan tampilan

Grafik merupakan hal yang penting dalam fisika komputasional. Pentingnya grafik dapat dibagi dalam 2 bagian besar, yaitu untuk analisis dan tampilan. Tampilan grafik sangat penting ketika kita menyajikan hasil yang sudah kita peroleh kepada masyarakat (baik umum maupun profesional). Kekuatan grafik dalam hal ini adalah sifatnya yang persuasif; karena itu grafik sebagai tampilan harus bagus. Penggunaan grafik untuk alat bantu analisis menuntut kualitas yang berbeda. Bentuk tampilan tidak harus bagus namun harus mudah untuk dipahami oleh pemakainya. Kita harus mempunyai kontrol yang cukup luas terhadap data yang digrafikkan itu.

Fortran 77 mempunyai beberapa *software* untuk menggambar grafik seperti misalnya PLPLOT, PSPLLOT, GNUPLOT. Misalnya di sini ditinjau PSPLLOT. Kita harus membuat dulu program untuk tampilan grafik. Program yang sudah kami susun (dalam Fortran 77) akan menjadi *template* untuk keperluan mendatang. Gambar grafik yang dihasilkan (bisa dilihat pada gambar 1) sangat bagus karena kita dapat mengatur semua segi tampilan secara terperinci. Namun tiap gambar harus menempati satu fail tersendiri sehingga kurang praktis dalam penggunaannya untuk analisis, lebih-lebih jika kita memerlukan banyak gambar. Ketika menganalisis keluaran program kita berhadapan dengan banyak jendela yang dibuka bergantian, tidak merupakan satu kesatuan.



Gambar 1. Grafik dengan PSPLLOT

Di sinilah pentingnya grafik mesin ketik. Jenis grafik ini berpenampilan sangat sederhana dan berkesan primitif namun cukup memenuhi syarat untuk analisis dan tampilannya bisa disatukan dengan keluaran program yang lain. Kita cukup menggunakan satu jendela untuk seluruh keluaran program (seperti penggunaan lembar kerja [worksheet] dalam Maple).

3.2. Program grafik mesin ketik dalam Fortran 77

Dalam makalah ini kami melaporkan penyusunan program sederhana (dalam Fortran 77) untuk menghasilkan grafik mesin ketik. Data yang masuk misalnya ($x, y = f(x)$) ditentukan daerah absisnya dengan menentukan nilai x maksimal ($= x_{max}$) dan x minimalnya ($= x_{min}$). Kemudian rentang $x_{max}-x_{min}$ dibagi menjadi 60 bagian dengan interpolasi. Kelemahan grafik mesin ketik memang berkurangnya resolusi. Daerah ordinat juga dibagi menjadi 30 bagian. Program selanjutnya hanya berusaha untuk menuliskan 60 nilai ordinat itu pada tempat yang sesuai berdasarkan tingkatannya (dimana disediakan 30 tingkatan) dan selanjutnya melengkapi gambar grafik dengan sumbu dan nilai absis-ordinat. Contoh hasilnya adalah sebagai berikut.

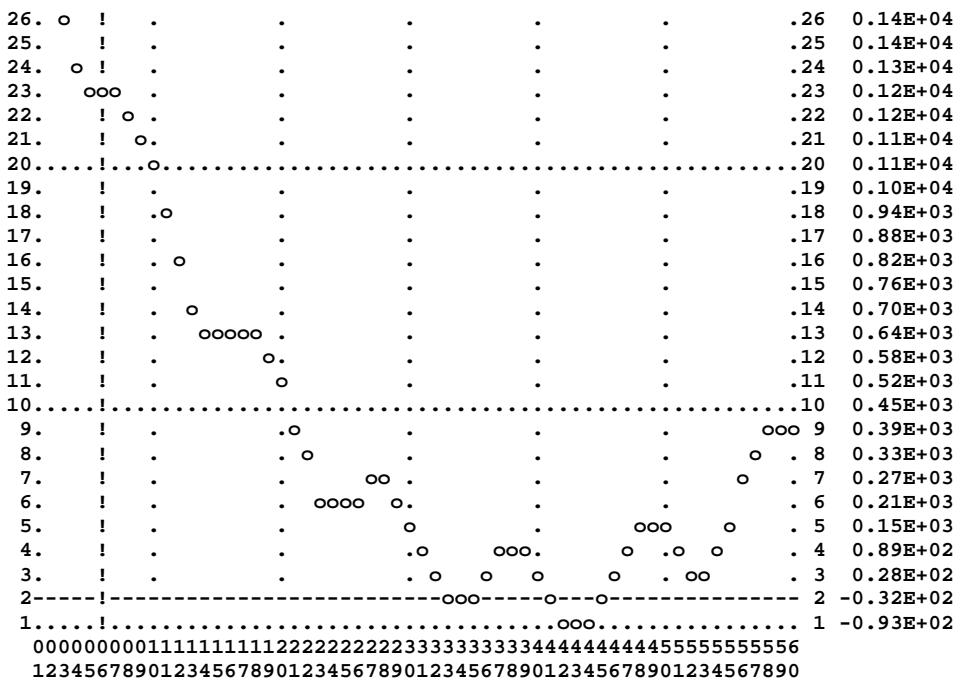
-----GRAFIK-----

$f(x) = (x-35)^2 + 100 * \sin(2 * \pi * x / 10)$

1	-0.550E+01	13	0.670E+01	25	0.189E+02	37	0.311E+02	49	0.433E+02
2	-0.448E+01	14	0.772E+01	26	0.199E+02	38	0.321E+02	50	0.443E+02
3	-0.347E+01	15	0.874E+01	27	0.209E+02	39	0.331E+02	51	0.453E+02
4	-0.245E+01	16	0.975E+01	28	0.220E+02	40	0.342E+02	52	0.464E+02
5	-0.143E+01	17	0.108E+02	29	0.230E+02	41	0.352E+02	53	0.474E+02
6	-0.415E+00	18	0.118E+02	30	0.240E+02	42	0.362E+02	54	0.484E+02
7	0.602E+00	19	0.128E+02	31	0.250E+02	43	0.372E+02	55	0.494E+02
8	0.162E+01	20	0.138E+02	32	0.260E+02	44	0.382E+02	56	0.504E+02
9	0.264E+01	21	0.148E+02	33	0.270E+02	45	0.392E+02	57	0.514E+02
10	0.365E+01	22	0.159E+02	34	0.281E+02	46	0.403E+02	58	0.525E+02
11	0.467E+01	23	0.169E+02	35	0.291E+02	47	0.413E+02	59	0.535E+02
12	0.569E+01	24	0.179E+02	36	0.301E+02	48	0.423E+02	60	0.545E+02

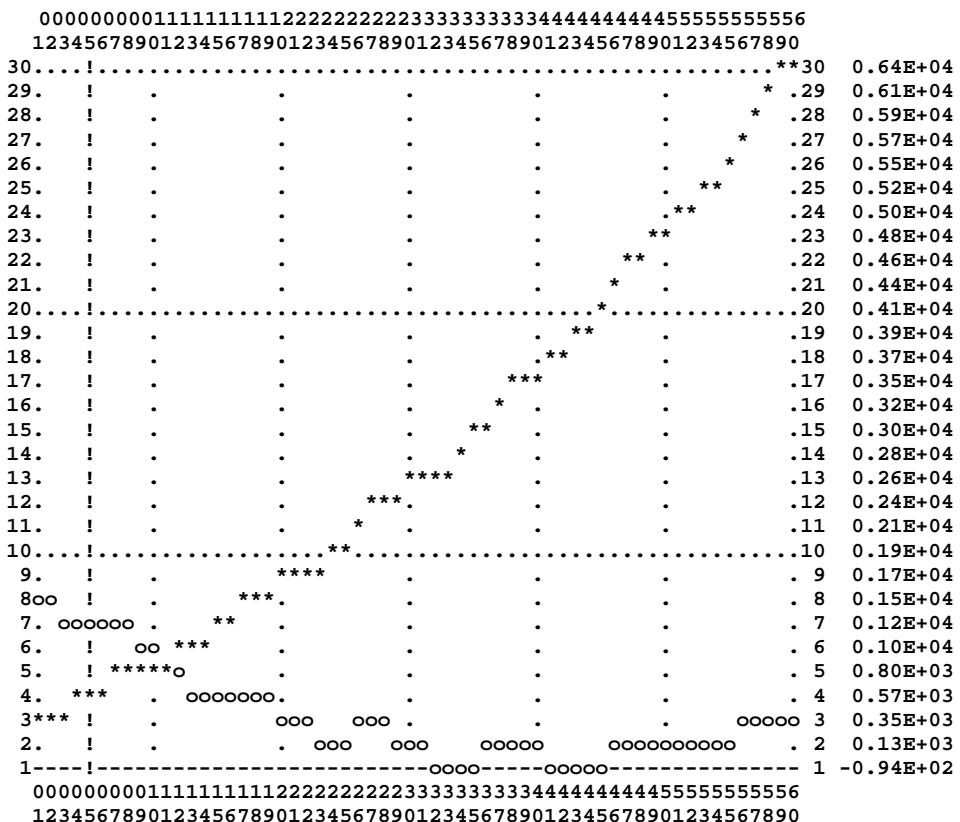
00000000011111111122222222333333334444444445555555556
12345678901234567890123456789012345678901234567890

30	0.17E+04
29.	0.16E+04
28.	0.15E+04
27.	0.15E+04



Gambar 2. Grafik mesin ketik

Kita juga bisa menggambarkan 2 grafik. Lebih dari 2 grafik akan sangat menyulitkan pengamatan sehingga tidak disarankan. Hasilnya adalah sebagai berikut.



Gambar 3. Grafik untuk 2 kurva

3.3. Contoh penggunaan

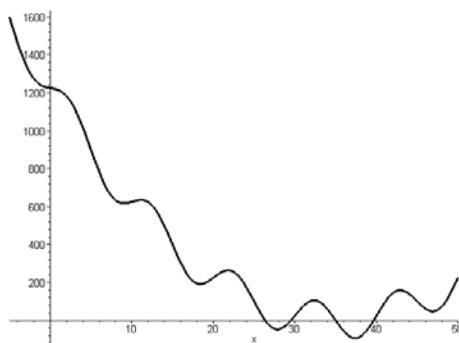
Sebagai contoh penggunaan adalah penyelesaian persamaan nonlinear. Dari gambar 2 kita bisa melihat bahwa ada beberapa akar di antara 30 (=24) dan 50 (=44,3). Dengan menggunakan metode *bisection* atau Newton-Raphson kita akan dapat menentukan akar-akar itu dengan ketelitian yang diinginkan. Analisis ini dapat dibandingkan dengan pemrograman Maple sebagai berikut.

```
> h:=(x-35)**2 + 100*sin(2.*Pi*x/10);
```

```
> fsolve(h,x);
```

26.34679796

```
> plot(h,x=-5..50,color=black,thickness=3);
```



4. Kesimpulan

Kami sudah berhasil menyusun program (yang selanjutnya bisa digunakan sebagai *template*) dalam Fortran g77 beserta PSPLLOT untuk menghasilkan grafik berpenampilan bagus. Kami juga sudah berhasil menyusun program (juga dalam Fortran g77) untuk menghasilkan grafik mesin ketik yang berpenampilan primitif namun cukup bagus untuk digunakan dalam analisis. Kesimpulannya adalah bahwa dengan tambahan perlengkapan ini, maka Fortran g77 yang dapat diunduh secara bebas lewat internet (merupakan *open source*) bersama *library* numerik SLATEC (yang sangat ekstensif dan sudah teruji handal) yang juga bebas, meskipun sudah cukup tua, masih tetap merupakan peralatan handal dalam fisika komputasional.

Daftar acuan

- Date, A.W., 2005, *Introduction to Computational Fluid Dynamics*, Cambridge University Press, Cambridge, UK
- Decyk, V.K., 1995, Skeleton PIC Codes for Parallel Computers, *Computer Physics Communications*, **87**, 87-94
- Devlin, K., 2000, *The Language of Mathematics*, W.H. Freeman and Company, USA
- Langtangen, H.P., 2004, *A Quick Fortran 77 Course*, University of Oslo, Norway, [http : heim.ifi.uio.no/~hpl/INF5660/F77.pdf](http://heim.ifi.uio.no/~hpl/INF5660/F77.pdf)
- Miller, J.A., 2004, *Promoting Computer Literacy through Programming Python*, disertasi Doktor, University of Michigan, USA

---oOo---