



KEMENTERIAN
PENDIDIKAN DAN KEBUDAYAAN



POLITEKNIK
NEGERI JEMBER

Algoritma Struktur Data

Modularity and Data Abstraction

Ulfa Emi Rahmawati, S.Kom., M.Kom.

Modularity

- Menggambarkan sebuah program yang diorganisasikan ke dalam modul-modul yang digabungkan secara longgar (*loosely*) dan sangat kohesif”
- “Sebuah teknik yang menjaga kompleksitas dari sebuah program besar yang dapat dikelola dengan secara sistematis mengontrol interaksi dari komponen-komponennya”

Secara Singkat:

- Modularity adalah sejauh mana komponen sistem dapat dipisahkan dan digabungkan kembali
- Disini kita akan mengetahui tentang Abstract Data Type (ADT)
- Untuk lebih memahami tentang ADT, mari kita lihat ilustrasinya

Sistem Informasi Siswa

- Misalkan kita akan membuat Sistem informasi untuk menyimpan data siswa
- Ada 4 menu:
 - ✓ Tambahkan data baru
 - ✓ Hapus data
 - ✓ Edit data
 - ✓ Lihat data yang disimpan



Apa yang biasanya kita lakukan?

Basic Algorithm Style

Type student < ...
>

Dictionary

.....

function add_student(...)

.....

Algorithm

.....

.....

Function add_student(...)

.....

Function delete_student(...)

.....

Procedure

.....

Type description of student, name, id, class, etc.

Declaring Variables and name of functions that will be used in main program

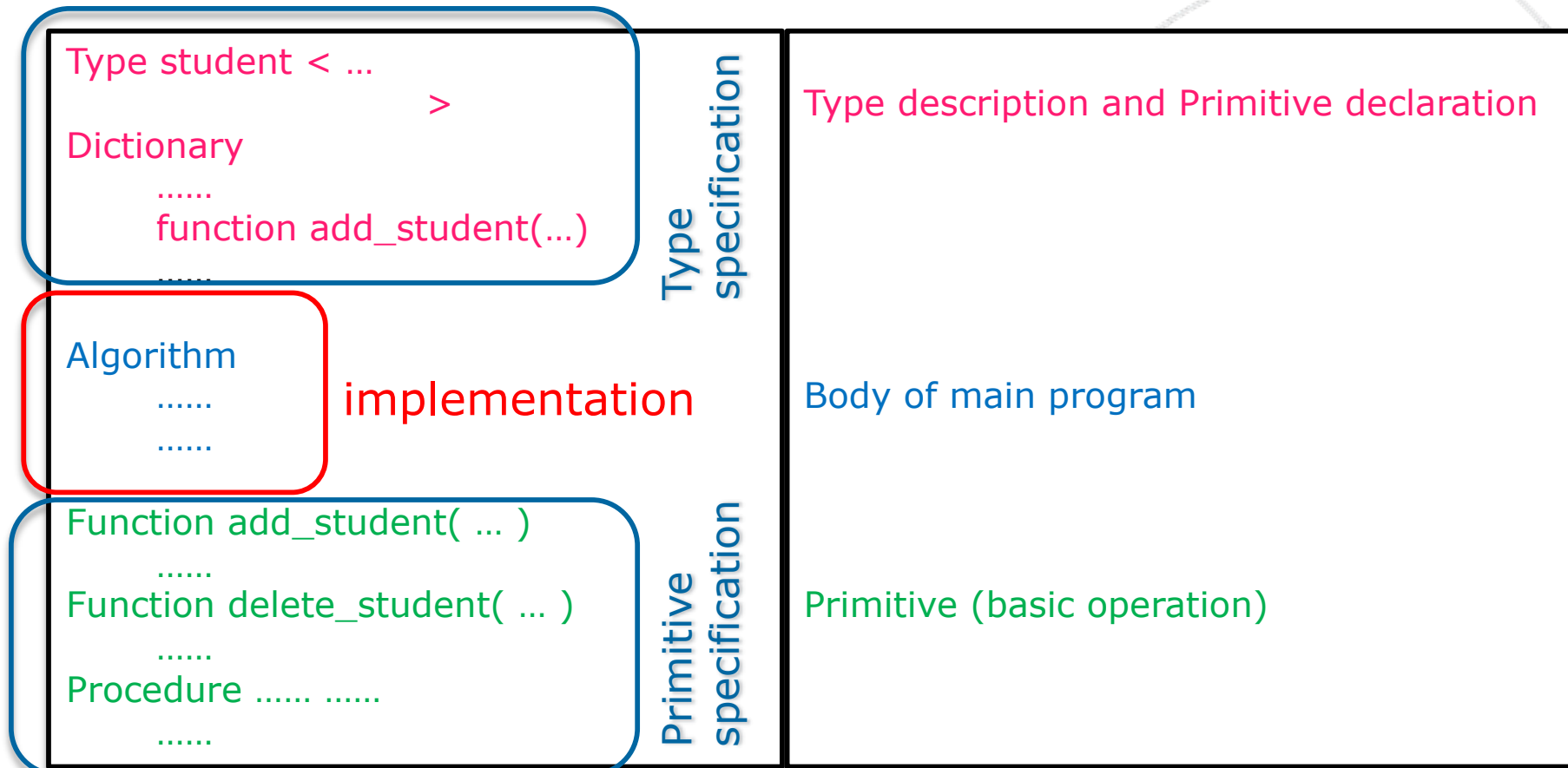
Main program, contains menu, interface, etc.
The main program will use the functions available or has been specified

Function specification, basic operation for student

All in one file (one script)

Sebenarnya kita punya

3 parts of program or 3 modules



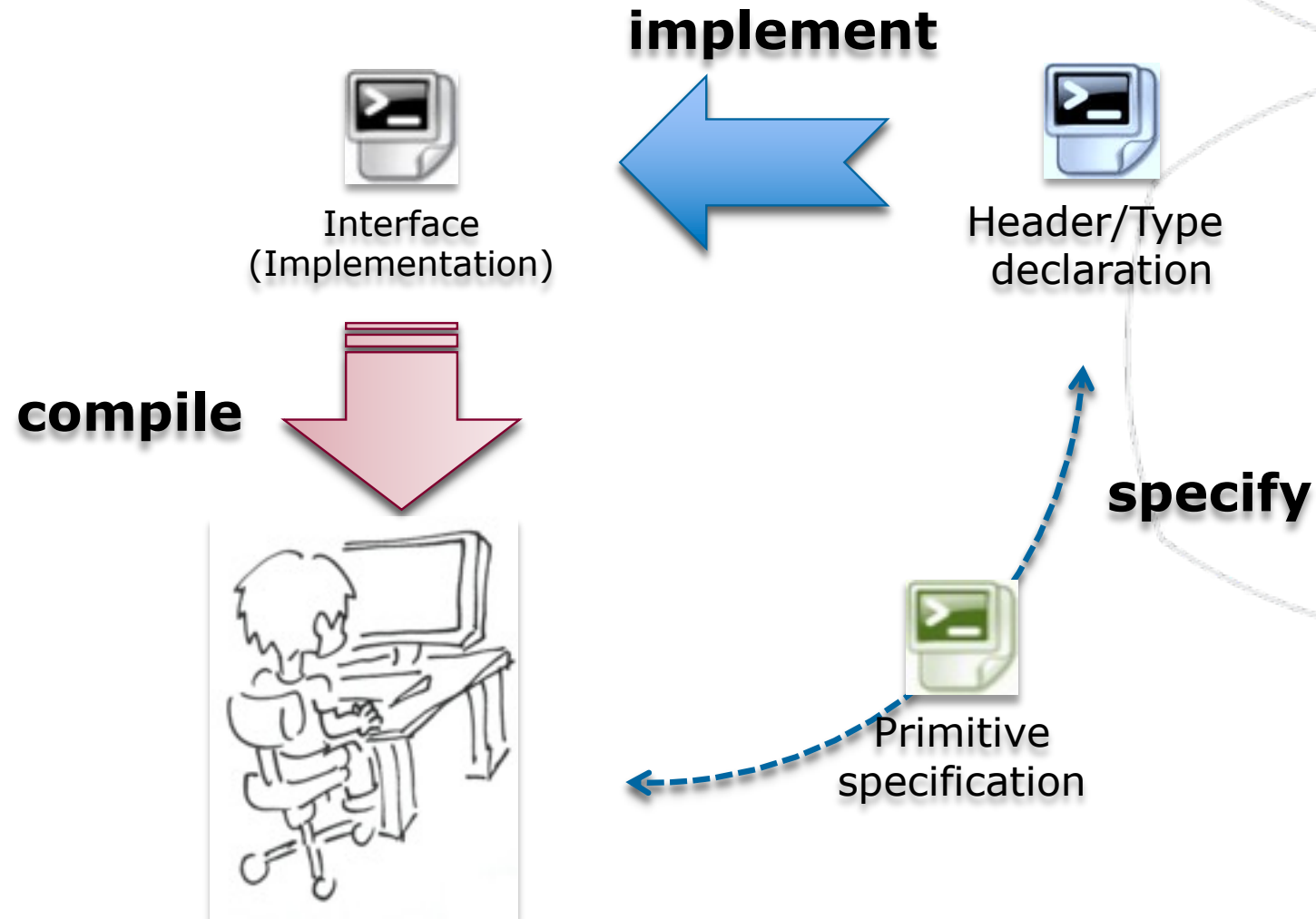
Abstract Data Type

- Tujuan ADT adalah untuk memisahkan antara spesifikasi dan implementasi
- ADT sendiri adalah bagian Spesifikasi yang berisi deklarasi TYPE dan spesifikasi PRIMITIF

Dengan demikian kita akan memiliki ADT style Algorithm

Basic Algorithm Style : 1 file	ADT Style : At least 3 files
Type student < ... Dictionary function add_student(...)	Header / type declaration Type student < function add_student(...) function delete_student(...)
Algorithm	Primitive specification Function add_student(...) Function delete_student(...)
Function add_student(...) Function delete_student(...) Procedure	Main / body program Algorithm

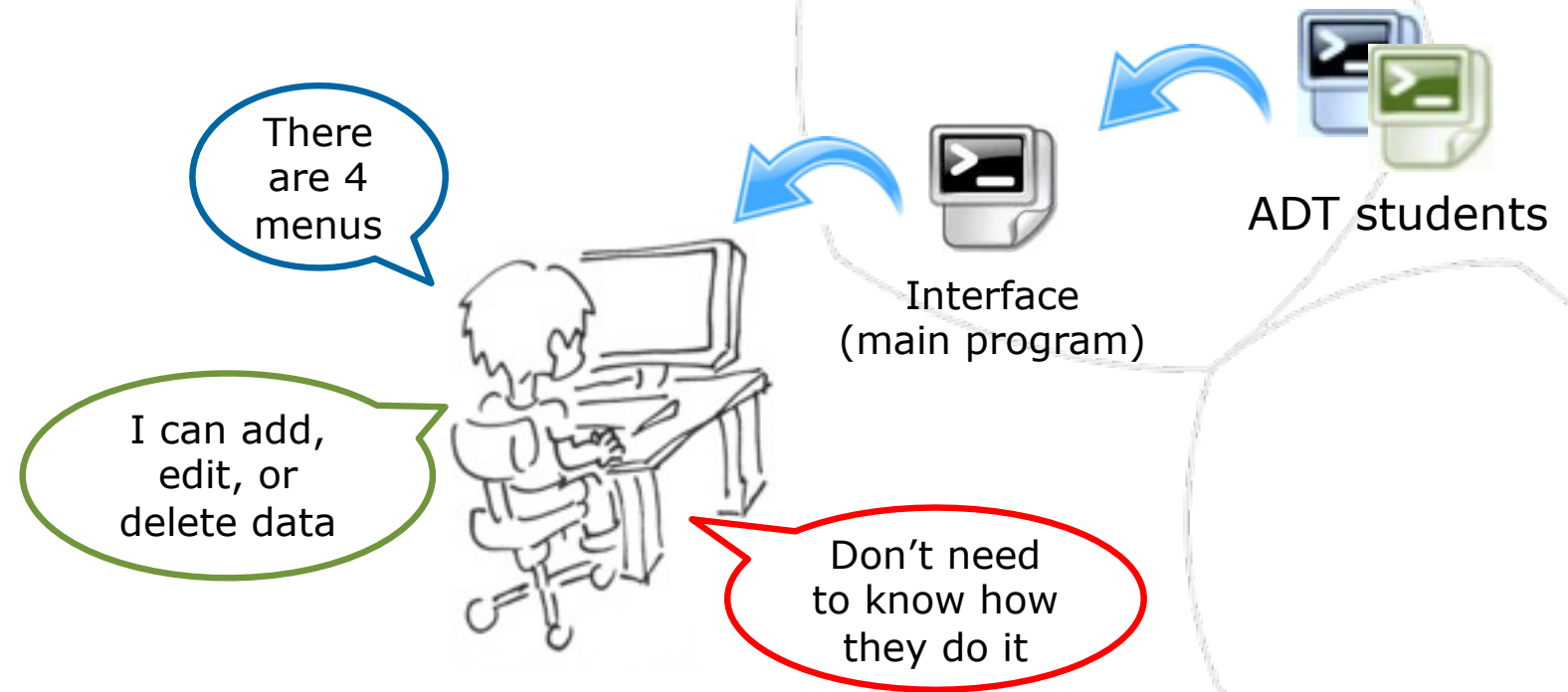
Konektivitas Antar Modul



Mengapa kita membutuhkan ADT?

Keamanan (*Security*)

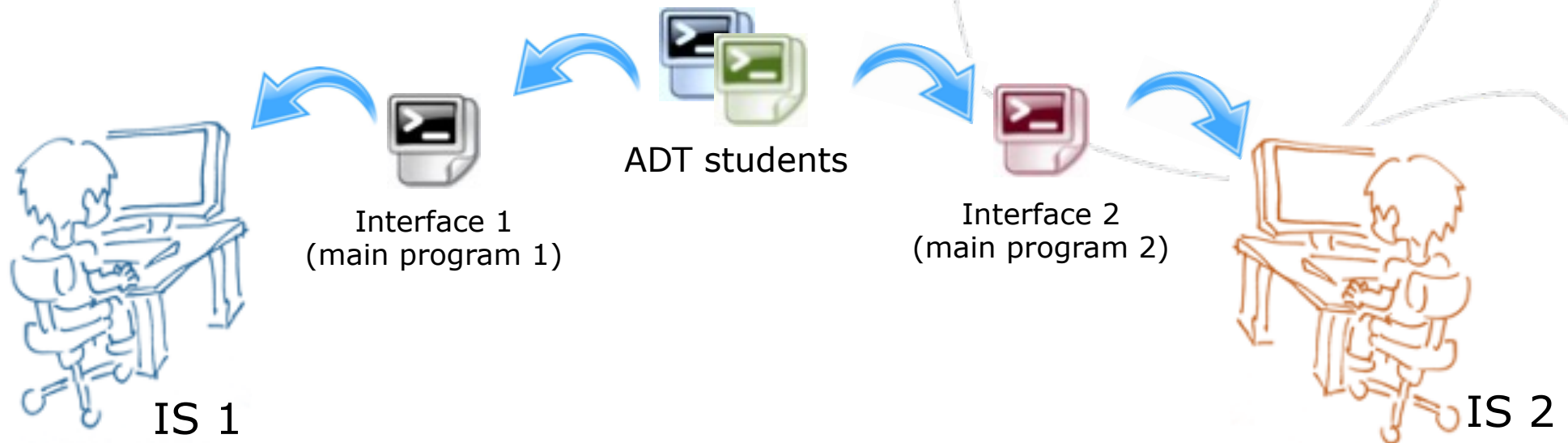
- ✓ Pengguna hanya perlu mengetahui spesifikasi fitur tanpa perlu diberikan implementasi detail dari fitur tersebut.



Mengapa kita membutuhkan ADT?

Dapat digunakan kembali (*Reusability*)

- ✓ Misalkan kita akan membangun sistem informasi lain yang juga menggunakan data siswa (tambah, edit, hapus)
- ✓ Kita tidak perlu mengkodekan ADT lagi, kita bisa menggunakan yang sudah ada





**POLITEKNIK
NEGERI JEMBER**

Thank You

Politeknik Negeri Jember

Jalan Mastrip Kotak Pos 164 Jember 68101

Telp. (0331) 333532-34

Fax. (0331) 333531

