# History of creation of IDEF0

During the 1970s, the United States Air Force Program for Integrated Computer Aided Manufacturing (ICAM) sought to increase manufacturing productivity through systematic application of computer technology.

To get this purpose achieved, the Structured Analysis and Design Technique (SADT) has been developed and field-tested during the period of 1969 to 1973 by Douglas T. Ross and SofTech. Inc.

The United States Air Force commissioned the developers of SADT to develop a function modeling method for analyzing and communicating the functional perspective of a system.

The ICAM program developed a series of techniques known as the IDEF (ICAM Definition) techniques

# Classification of IDEF models

IDEF1, used to produce an "information model”

IDEF1x, used to produce "data model"

IDEF2, used to produce a "dynamics model"

IDEF3, used to produce "process model"

IDEF4, used to produce object-oriented design

IDEF5, used to produce ontology description capture

# IDEF0 as official document

In 1993 the government of the USA accepted this standard as a federal standard.

In 2000 the new version of ISO 9000:2000 standards was accepted by International Organization of Standardization

IDEF0 used as the main document for standardization in Russian Federation In 2000.

Ukraine use this standard

# IDEF0 has the following characteristics:

•*Generic*. It is comprehensive and expressive, capable of graphically representing a wide variety of business, manufacturing and other types of enterprise operations to any level of detail, so for analysis of systems and subject areas of varying purpose, scope and complexity

•*Rigorous and precise*. It is a coherent and simple language, providing for rigorous and precise expression, and promoting consistency of usage and interpretation.

•*Concise*. It enhances communication between systems analysts, developers and users through ease of learning and its emphasis on hierarchical exposition of detail.

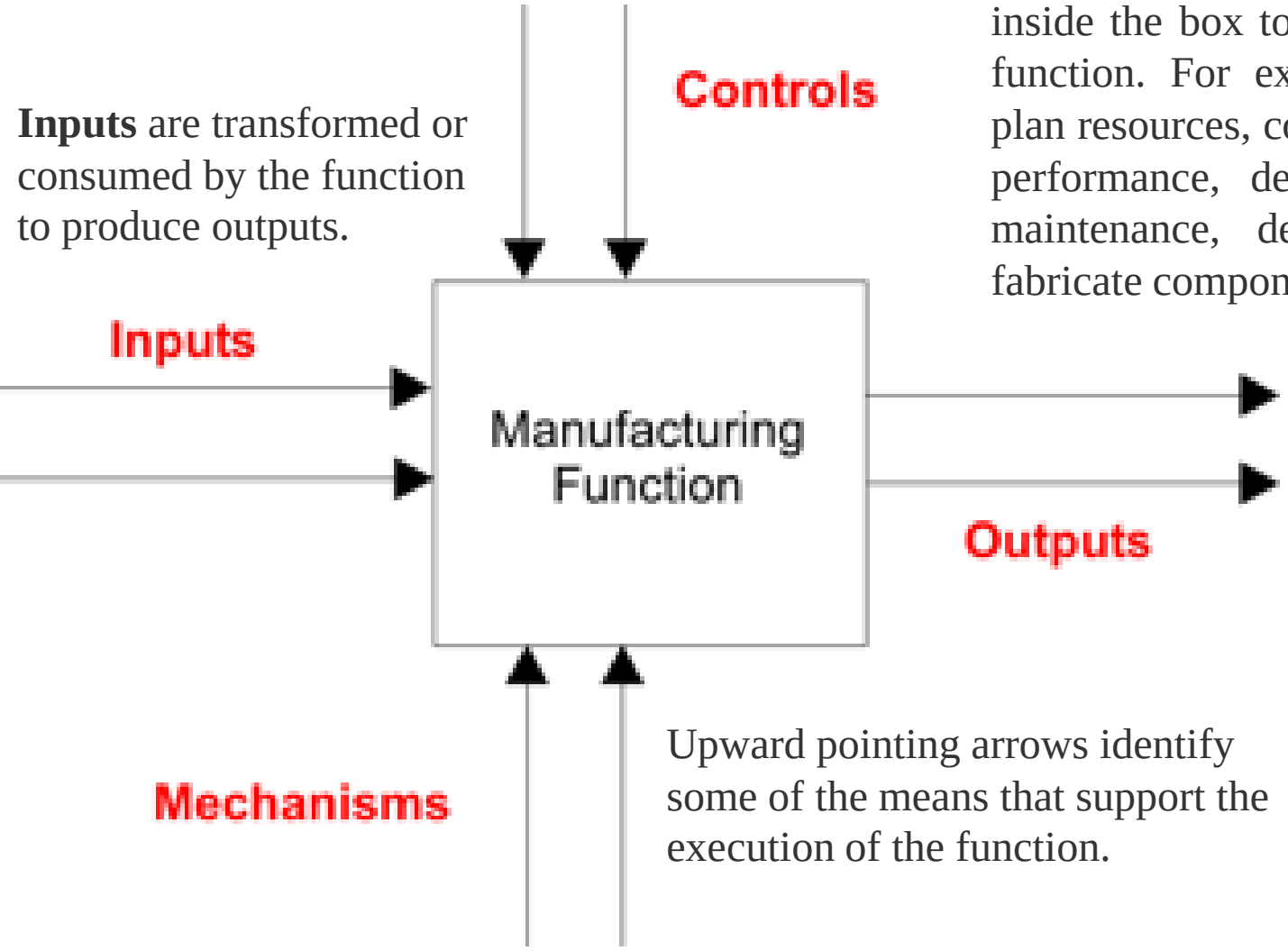
•*Conceptual*. It can be used for representation of functional requirements independent of physical or organizational implementations. It is well-tested and proven, through many years of use in Air Force and other government development projects, and by private industry.

•*Flexible*. It can be supported several phases of the life cycle of a project It can be generated by a variety of computer graphics tools; numerous commercial products specifically support development and analysis of IDEF0 diagrams and models.

# Structure of IDEF0 scheme

**Controls** specify the conditions required for the function to produce correct outputs.

**A box** provides a description of what happens in a designated function or activity. The verb or verb phrase placed inside the box to describe the modeled function. For example: process parts, plan resources, conduct review, monitor performance, design system, provide maintenance, develop detail design, fabricate component, inspect part.



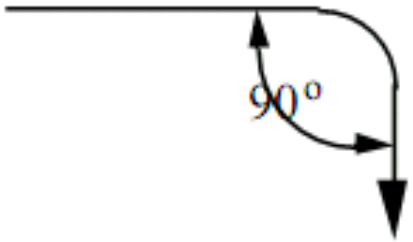

**Inputs** are transformed or consumed by the function to produce outputs.

**Outputs** are the data or objects produced by the function.

Upward pointing arrows identify some of the means that support the execution of the function.

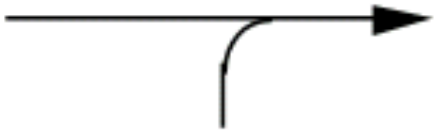
Different types of arrows

- Straight line arrow segment



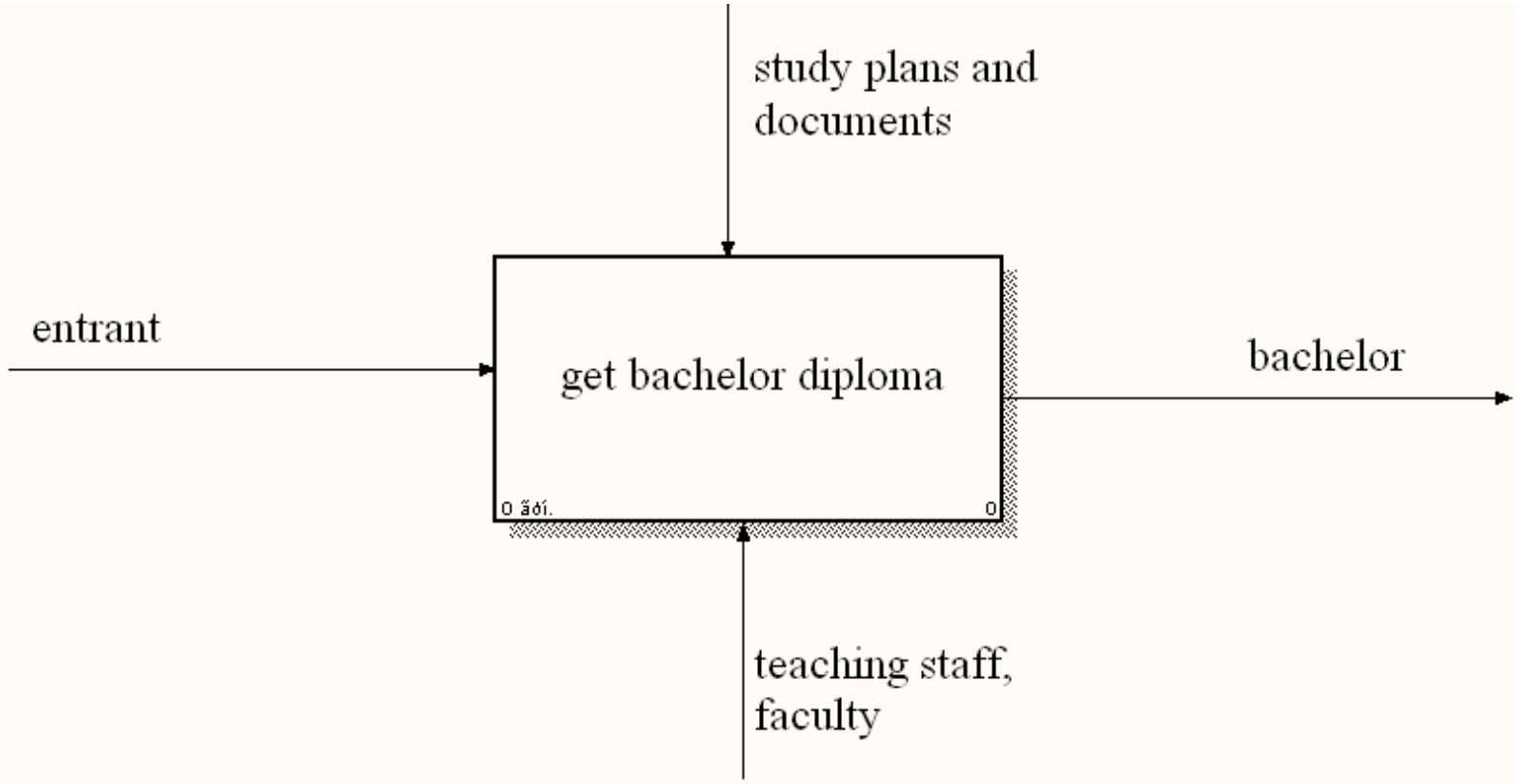

- Curved arrow segment; corners are rounded with 90 degree arcs

- Forking arrows

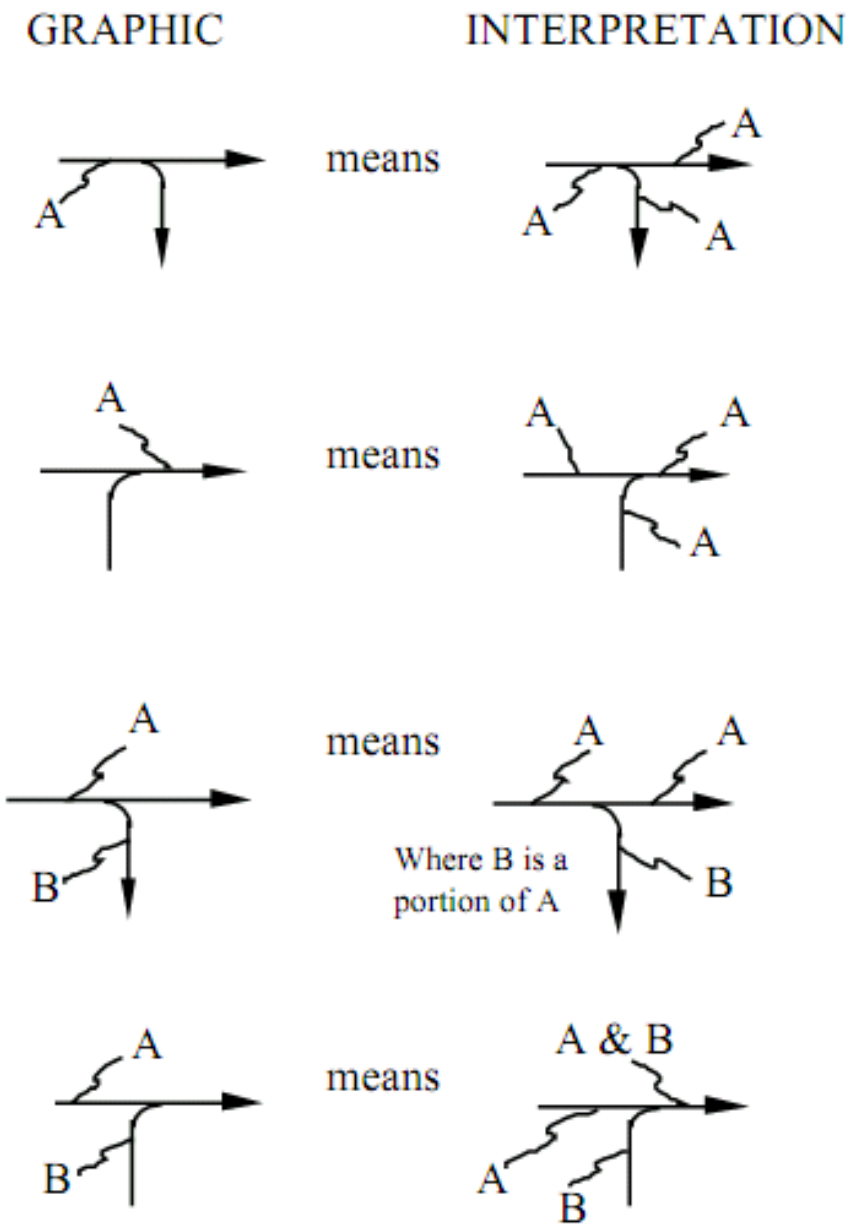


- Joining arrows

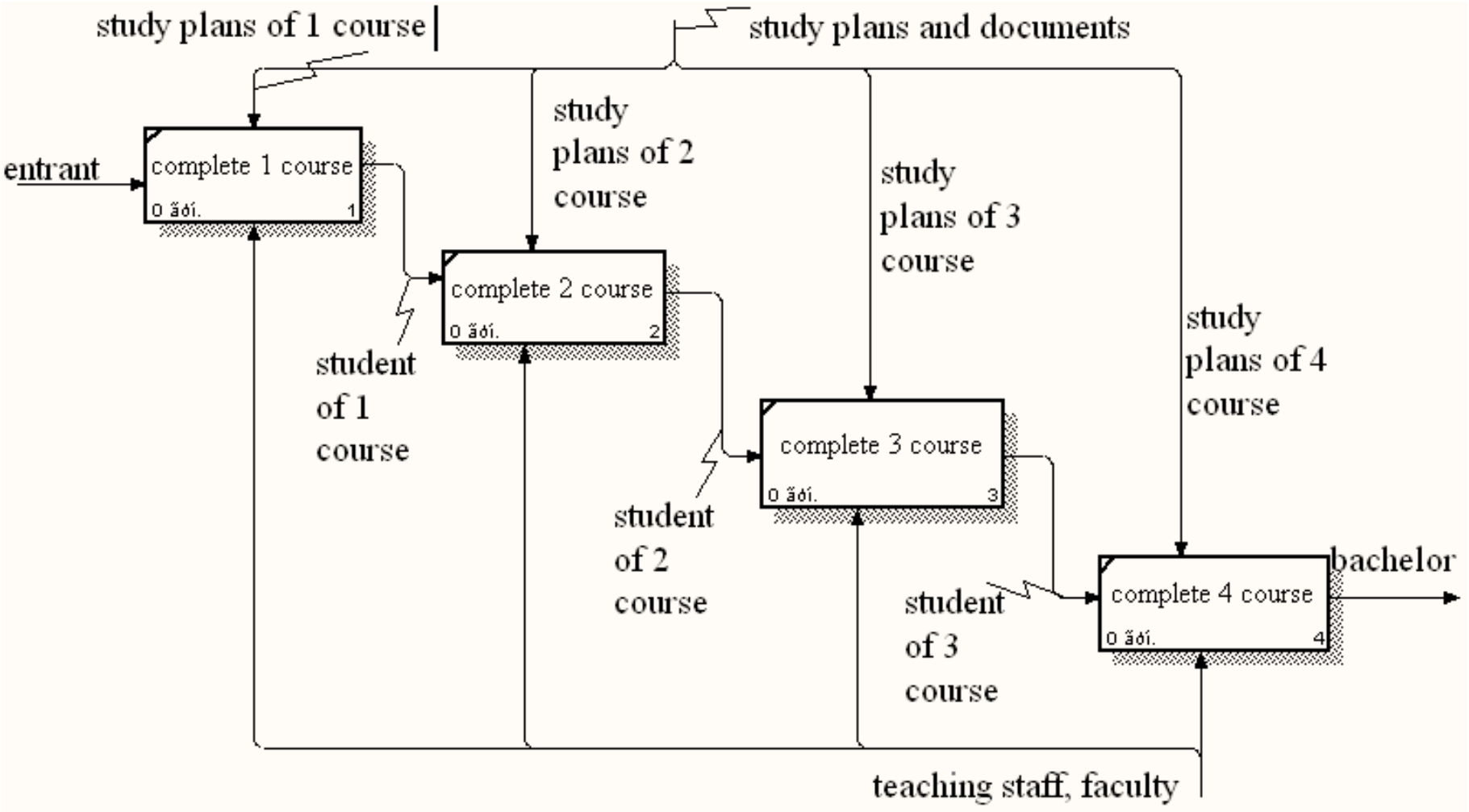
Top-level for process “Get bachelor diploma”



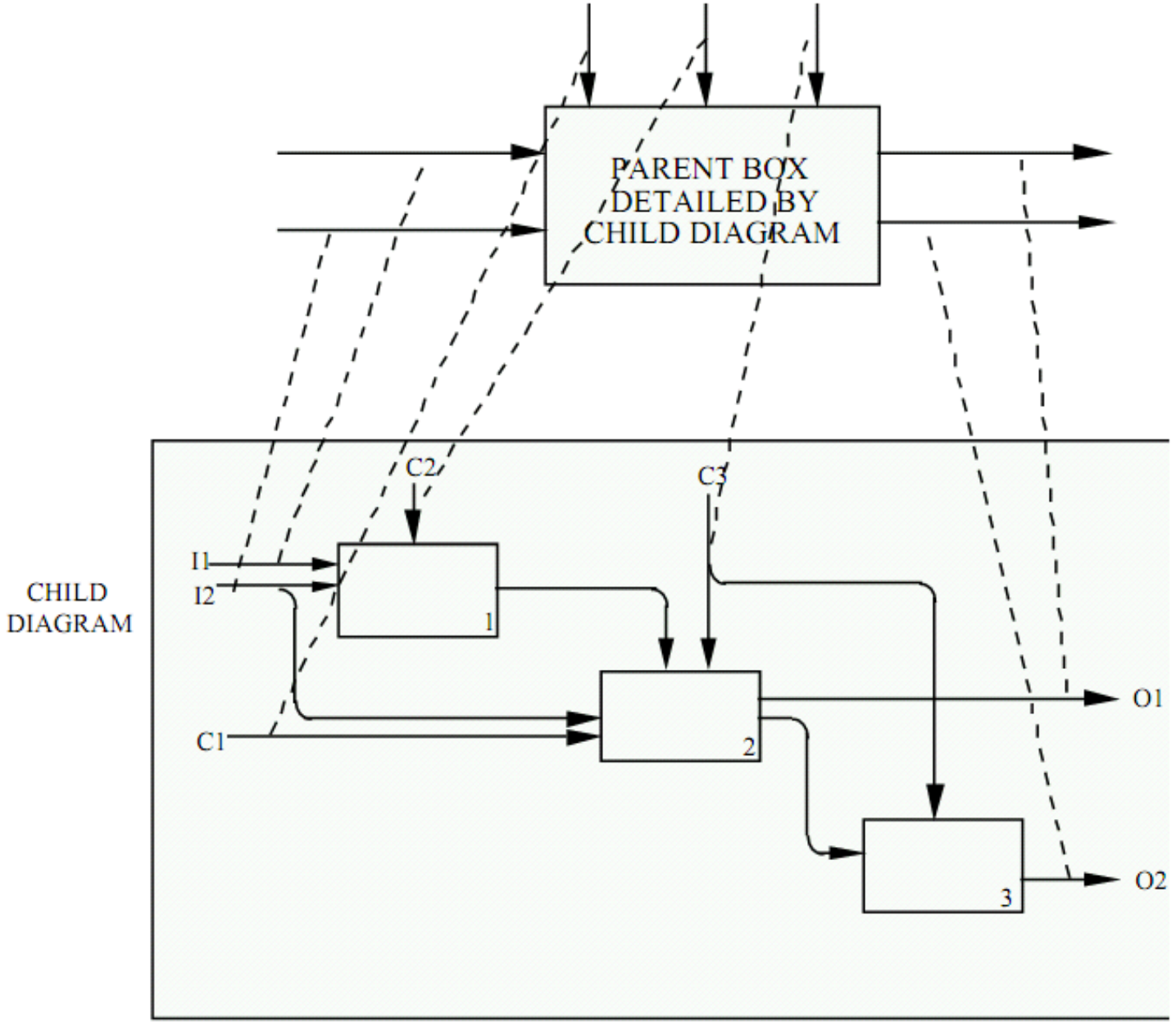
Arrow may have branch (fork or join)



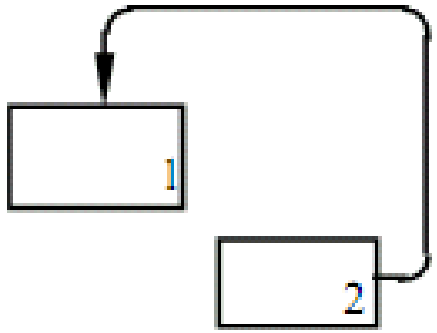
Example of branch



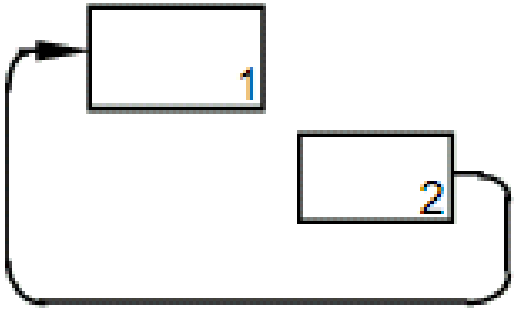
There are cases when the arrow roles are changed on parent and child diagram.



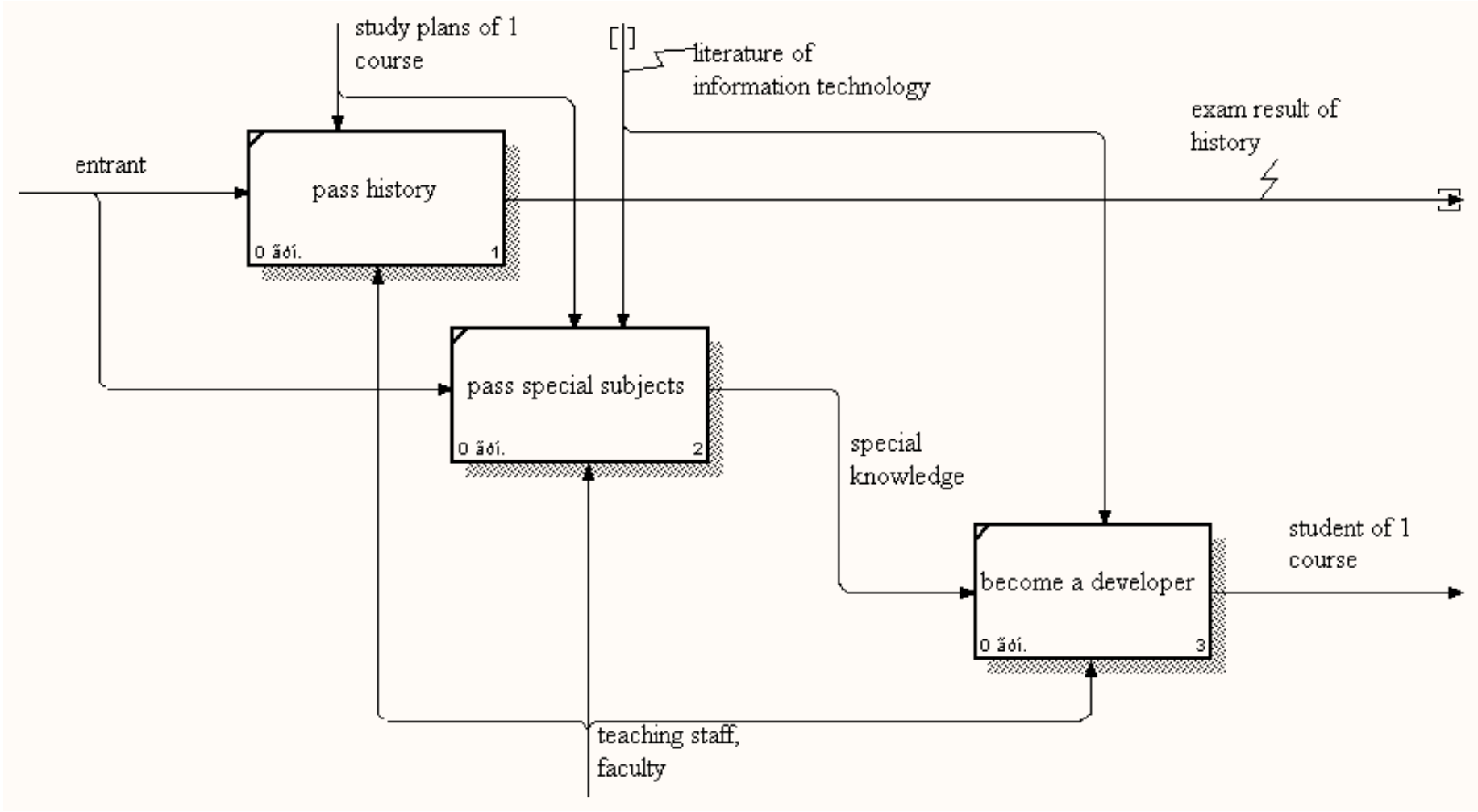
Control feedbacks shall be shown as “up and over”.



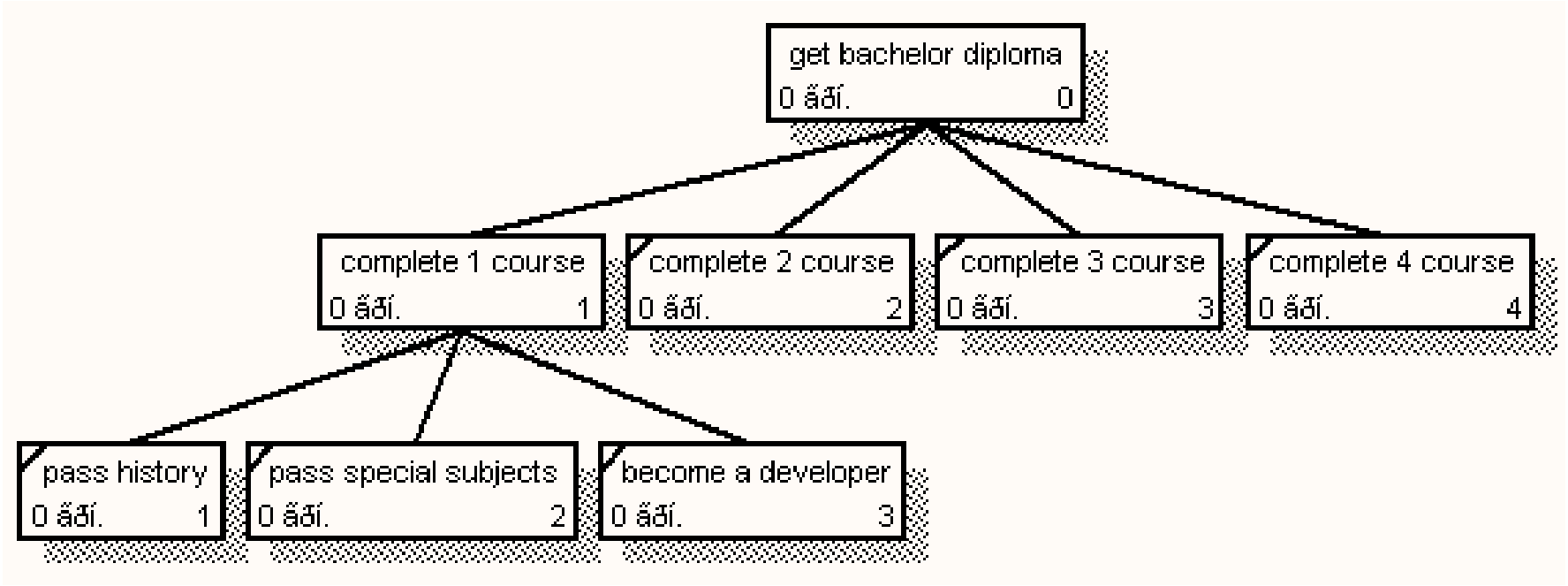

Input feedbacks shall be shown as “down and under”.



# Example of a tunneled arrow



# Node tree


get bachelor diploma
0 ãðí. 0
complete 1 course
0 ãðí. 1
complete 2 course
0 ãðí. 2
complete 3 course
0 ãðí. 3
complete 4 course
0 ãðí. 4
pass history
0 ãðí. 1
pass special subjects
0 ãðí. 2
become a developer
0 ãðí. 3

# Task:

create as-is and to-be models for “settle into NTU KhPI hostel”