**2 B) Co to je algoritmus, jaké jsou jeho vlastnosti? Objasněte schematické vyjádření algoritmu vývojovým diagramem, symboly vývojového diagramu, zápis algoritmu v programovacím jazyku. Uveďte jednoduchý příklad = nakreslete vývojový diagram a proveďte zápis téhož algoritmu v programovacím jazyku Python.**

**Algoritmus**

Algoritmus je konečná, přesně definovaná a uskutečnitelná posloupnost kroků, která převádí zadané vstupy na požadované výstupy.

**Vlastnost Význam**

Konečnost Po konečném počtu kroků vždy skončí.

Jednoznačnost (determinismus) Každý krok je zadaný tak, že jej lze provést jen jedním způsobem.

Hromadnost (obecnost) Řeší celou třídu úloh stejného typu, ne jen jeden konkrétní vstup.

Vstup / výstup Má definované nula či více vstupních a alespoň jeden výstupní údaj.

Efektivita Provádění je realizovatelné s reálně dostupnými prostředky (čas, paměť).

**Schematické vyjádření – vývojový diagram**

Flowchart zobrazuje kroky algoritmu graficky; **tok řízení** sledují orientované šipky.

Vývojový diagram slouží ke komunikaci s neprogramátory, k revizi logiky a k dokumentaci procesu před samotným kódováním.

| **Symbol** | **Význam (nejčastější použití)** |
| --- | --- |
| 🞅 **Ovál** | **Start / End** procesu |
| ▱ **Paralelogram** | Vstup nebo výstup dat |
| □ **Obdélník** | Zpracování (výpočet, přiřazení) |
| ◇ **Kosočtverec** | Rozhodnutí (podmínka s větvením „ano/ne“) |
| 🔀 **Konektor** | Přeskočení mezi částmi složitého diagramu |
| → **Šipka** | Řídicí tok |

**Zápis algoritmu v programovacím jazyce**

Jakmile je logika jasná, **převede se do syntaxe konkrétního jazyka** (např. Python, C#, Java).  
Převod není mechanický – programátor musí:

1. definovat datové typy a proměnné,
2. převést rozhodovací uzly na podmíněné příkazy (if, while, …),
3. ošetřit vstupy/výstupy a chybové stavy,
4. dodržet idiomatický styl jazyka.

**Příklad**

Čte dvě čísla z jednoho řádku a vytiskne to větší. ​

Python

a, b = map(int, input().split()); print(max(a, b))