

Adatbázis és szoftverfejlesztés elmélet

Témakör 1.

Összefoglalás és kérdések

Program:

A Számítógép számára érthető utasítássorozat.

Programozás:

Szűkebb értelemben a kódolást értjük ezalatt. **Tágabb értelemben** hétköznapi és speciális feladatok modellezése a számítógépen. A feladat kidolgozásától kezdődően a kész program leírásáig minden lépés ide tartozik.

A Neumann-elvek:

1. A számítógép legyen soros működésű:

A gép az egyes utasításokat egymás után, egyenként hajtja végre

2. A számítógép a kettes számrendszert használja, és legyen teljesen elektronikus:

A kettes számrendszert és a rajta értelmezett aritmetikai, ill. logikai műveleteket könnyű megvalósítani kétállapotú áramkörökkel (pl.: 1- magasabb feszültség, 0 - alacsonyabb feszültség)

3. A számítógépnek legyen belső memóriája:

A számítógép gyors működése miatt nincs lehetőség arra, hogy minden egyes lépés után a kezelő beavatkozzon a számítás menetébe. A belső memóriában tárolhatók az adatok és az egyes számítások részeredményei, így a gép bizonyos műveletsorokat automatikusan el tud végezni.

4. A tárolt program elve:

A programot alkotó utasítások kifejezhetők számokkal, azaz adatként kezelhetők. Ezek a belső memóriában tárolhatók, mint bármelyik más adat. Ezáltal a számítógép önállóan képes működni, hiszen az adatokat és az utasításokat egyaránt a memóriából veszi elő.

5. A számítógép legyen univerzális:

A számítógép különféle feladatainak elvégzéséhez nem kell speciális berendezéseket készíteni. Ugyanis, Turing angol matematikus bebizonyította, hogy az olyan gép, amely el tud végezni néhány alapvető műveletet, akkor az elvileg bármilyen számítás elvégzésére is alkalmas.

Programozási nyelvek fejlődése (generációk):

1. GL (alacsony szintű nyelvek)

Az első programozási nyelv a gépi kód volt. Ennek a nyelvnek az utasításait a számítógép képes volt közvetlenül, minden átalakítás nélkül végrehajtani, értelmezni. A problémákat nagyon precízen kellett megfogalmazni, lépésekre bontani. Emellett a nyelv erősen gépfüggő volt, hiszen minden gépen más és más utasításokat használt, az adott problémát minden géptípus

esetén másképpen kellett leírni, megfogalmazni, alkalmazkodva a számológység sajátosságaihoz. Hátrányai mellett néhány előnyt is meg kell említenünk:

- A leggyorsabb programot eredményezi, mivel nem tartalmaz felesleges utasításokat.
- Az egyes utasítások egységes (egy vagy kétcímes) formában vannak leírva. pl: OBOA OCOE OD00 – az első négy szám egy utasítás, a többi memóriacímek

A gépi kódú lehetőségekhez képest komoly előrelépést jelentett az Assembly nyelvek megjelenése, amelyek tulajdonképpen segédeszközt jelentettek a gépi kódú utasítások egyszerűbb megfogalmazásában. Az egyes gépi kódú utasításokhoz egy-egy *mnemonikus* kódot rendeltek hozzá, a tárcímeket pedig a memória kezdetéhez viszonyított relatív címekkel számították. Szintén újdonságnak számított, hogy az egyes memóriacímeket egy-egy szimbolikus névvel lehetett helyettesíteni.

- Minimális szókészlet (ma már csak hardver szintű vezérléshez használatos) pl: ADD, X, Y – az ADD utasítás, az X és Y memóriacímek

Az Assembly nyelv a gépi kódú programozás után igen nagy előrelépést jelentett, általa sokkal kényelmesebbé vált a programfejlesztési munka. Napjainkra az alacsony szintű programozási nyelvek háttérbe szorultak. Az Assembly-t ma már csak olyan feladatok esetén használják, ahol hardver szintű vezérlést kell megvalósítani.

2. GL (magas szintű nyelvek)

A számítógépek alkalmazási területének bővülése szükségessé tette a programok fejlesztési idejének csökkentését, azaz felmerült az igény, hogy a programokat minél gyorsabban írják meg a programozók. A gépi kód és az assembly nehézsége, géphez igazodása miatt nem volt erre alkalmas. A generáció jellemzői:

- A problémához igazodik, nem a számítógép tulajdonságaihoz.
- 1:N típusú programozás (1 utasítás több gépi szintű utasítást tartalmaz)
- Forráskód áttekinthetőbb

Nyelvek:

FORTRAN - Az első magas szintű programnyelv. Több változáson esett át, de ma is használatos. Főként matematikai és különböző tudományos területen.

ALGOL - A Pascal nyelv alapja A kutatásban és oktatásban használták.

COBOL - Több változáson esett át, de ma is használatos, főként az adatfeldolgozás területén.

BASIC - Egyszerűsége miatt az oktatásban volt használatos és a kezdő programozók is ezt használták.

3. GL

A második generáció programozási nyelvei már lényegesen függetlenebbek voltak a gépi logikától, mint az első generációs nyelvek, az elkészített programokat többféle gépen is használni lehetett. Azonban a nyelvek többsége egy-egy problémacsoport megoldására specializálódott, ami a programok fejlesztése, javítása során igen komoly akadályokat jelentett. Ezért:

- Megjelenik a strukturáltság mind módszerek, mind adatok tekintetében.
- Általános célú nyelvek megjelenése

Nyelvek:

MODULA-2 - Moduláris programozás, feladatok részekre bontása.

ADA - Főleg rendszerprogramozásra és gazdasági számítások területén.

PASCAL - Általános célú programozási nyelv.

C - Eredetileg UNIX rendszerprogramozásra találták ki, később elterjedt.

4. GL

Az 1980-as években kezdett széleskörűen elterjedni egy új szemléletű programozási technika, az objektum-orientált programozás, amely az elődeinél egyszerűbbé, hatékonyabbá, biztonságosabbá teszi a programkészítési munkát. Ez a programozási technika sokkal közelebb áll az emberi gondolkodáshoz, mint a *procedurális* programozás. Tehát jellemzők:

- Közelítés az emberi gondolkodáshoz
- Objektumorientált szemlélet

Nyelvek:

TURBO PASCAL - Pascal nyelv objektum-orientált lehetőségekkel.

Borland C++ - A C nyelv objektum-orientált lehetőségekkel bővített változata.

DELPHI - A Turbo Pascal nyelv továbbfejlesztése, vizuális környezet biztosítása.

JAVA - Általános célú, objektum-orientált nyelv C és C++ alapokon.

5. GL

A XX. század második felében a számítástechnika forradalmi változáson ment keresztül. A Neumann elvek megjelenése és az első memóriában tárolt gépi nyelvű program lefuttatása után alig ötven évvel már információs rendszerek, központilag vezérelt számítógép hálózatok programozása, döntéshozást támogató rendszerek fejlesztése a célunk. Nyilvánvaló, hogy ilyen jellegű munkák elvégzésére a hagyományos programozási nyelvek által biztosított eszközök nem megfelelőek. A hardver eszközök rohamos fejlődésére támaszkodva kezdett kialakulni a programozás során egy új szemlélet, mely az eddigi alapelvekkel szöges ellentétben állt.

- Párhuzamos programozás
- Hálózatok programozása

Kérdések:

1. Mit értünk program alatt?
2. Mit értünk programozás alatt?
3. Ismertesse röviden Neumann János számítógépekre vonatkozó elveit!
4. Mit jelent a tárolt program elve?
5. Sorolja fel a programozási nyelvek generációit, írjon le mindegyikhez két-két fontos jellemzőt!
6. Miért volt szükség magasabb szintű programozási nyelvek kifejlesztésére?