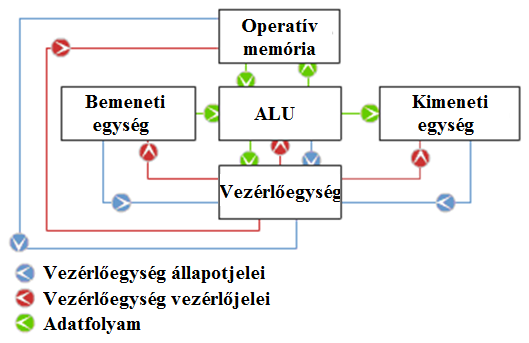
1. Személyi számítógépek

(Kidolgozta: Krascsenits Erik)

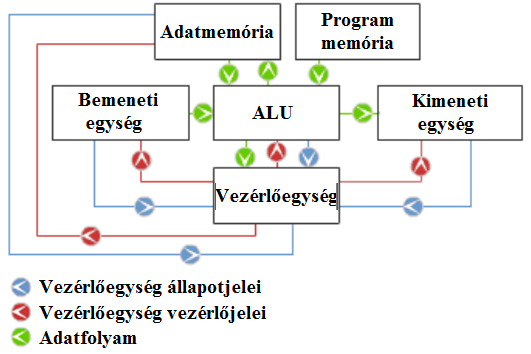
1. Tömbvázlat segítségével magyarázza el a Harvard architektúrájú számítógép működését és hasonlítsa össze a Neumann architektúrával.

**Neumann architektúra:**



A Neumann architektúrát Neumann János fejlesztette ki, 5 fő részből áll: operatív memória (futó programok, feldolgozott adatok helyezkednek el benne), ALU (aritmetikai és logikai számításokat végez), bemeneti egység (felhasználótól kér adatokat), kimeneti egység (felhasználóval közöl adatokat), vezérlőegység (vezérli a számítógép részeit vezérlő- és állapotjelekkel).

**Harvard architektúra:**



Howard Aiken fejlesztette ki. Hasonló a Neumann architektúrához, viszont itt a memória két részre van osztva, az egyikben a programok a másikban pedig az adatok vannak elmentve.

1. Írja le az alaplap részeit (chipset, sínek, interface-k, ...)

Az alaplap egy többrétegű nyomtatott áramköri lap, amely a személyi számítógépet alkotó elemek elektronikai alkatrészeit tartalmazza.

Részei:

* Processzorfoglalat
* RAM memóriahelyek
* BIOS/UEFI ROM
* Chipkészlet/lapkakészlet – A számítógép működéséhez szükséges vezérlőket tartalmazza, biztosítja a memória, a processzor és a perifériák közötti kommunikációt. Chipset architektúrák: North/South bridge architektúra, AHA elosztó architektúra, Multi-Threaded I/O Link architektúra. A chipset határozza meg a felhasználható processzor típusát, a RAM memória típusát, üzemmódját és méretét. Részei:
  + North Bridge (Északi híd) – A processzorral, déli híddal, RAM-mal, videókártyával kommunikál.
  + South Bridge (Déli híd) – A gyors soros és párhuzamos interfészeket üzemelteti, interfész az alaplap rendszereszközeivel szempen (pl. valós idejű óra), egyéb komponensek interfésze (pl. COM, LPT).
* Sínek/buszok – Vezetőkészlet, amely a processzort köti össze a számítógép többi komponensével.
  + típusai: címbusz, adatbusz, vezérlőbusz, tápbusz
  + átvitel iránya: egyirányú, kétirányú
  + átvitel módja: párhuzamos, soros
  + elhelyezkedés: belső (processzor-memória), külső (Déli híd-perifériák)
  + bitszélesség: 8, 16, 32, 64, 128-bites
  + szinkronizáció típusa: szinkron (processzorütemtől függ), multimaster (több eszköz vezérli), pszeudoszinkron (speciális jellel való késleltetés)
  + Fő buszok: FSB (CPU cache-North bridge), Memóriabusz (CPU cache-RAM), I/O buszok (PCI, PCIe, AGP)
* Interfészek (Alaplak külső csatlakozói) – RJ45, soros port, párhuzamos port, lila PS/2- billentyűzet, zöld PS/2 – egér, audio portok, USB port, Firewire, E-Sata, Thunderbolt.
* Alaplap belső elemei és csatlakozói: ventillátor és állapotjelző LED-ek csatlakozói, feszültségkonvertálók, hangkártya csatolók, SATA, PATA, FLOPPY, M-2, NVM2

1. Magyarázza el az operációs rendszerrel kapcsolatos alapfogalmakat (definíció, tömbvázlat, rendszer eszközök, erőforrások).

**Operációs rendszer definíciója:**

Az operációs rendszer programerőforrások együttese, amely lehetővé teszi a technikai (HW – Hardware) és program (SW – Software) alkalmazáserőforrások felhasználását felhasználói szinten.

**Tömbvázlat:**

A képen képernyőkép látható

Ez egy automatikusan létrehozott leírás.

**Rendszereszközök:**

**OS mag (kernel) –** A rendszer végrehajtó egysége, amely a memóriában van elmentve és rezidens programként fut (háttérben dolgozó program).

**Boot loader –** Elsőként indul, feladata az OS memóriába töltése és az irányítás átadása az OS-nek (indítási folyamat).

**Parancsértelmező (interpreter) –** Régen Monitornak is nevezték, biztosítja a kommunikációt az OS és a felhasználó között.

**Illesztőprogram (driverek) –** A ki- és bemeneti eszközöket kiszolgáló programok.

**Rendszer erőforrások:**

Az a hardverkörnyezet, amely alatt dolgozik az OS.

**Processzor** – Meghatározott szabályok alapján az egyes folyamatok utasításait végrehajtó egység.

**Operatív memória** – Az összes bináris kód ideiglenes tárhelye, amelyekkel aktuálisan dolgozik a processzor.

**Perifériák** – Az összes belső és külső megcímezhető eszköz, amelyeknek nincs operatív memória jellege (a processzor a perifériát egyként vagy memóriaterület csoportjaként látja).

**Állományrendszer** – Fizikai eszközön megvalósított szabályrendszer, amelynek célja az egy egészként megjelenő kódok kompakt csoportjának (állományok) hosszútávú tárolása.

1. Írja le az operációs rendszer indulásának egyes fázisait (POST, BIOS, MBR, BR..).

1. lépés – POST (Power On Self Test) – A számítógép komponenseinek önellenőrzésének folyamata. A hibákat a rendszer sípkódokkal (beep code) jelzi.

2. lépés – A BIOS beolvassa a CMOS memóriában tárolt konfigurációs beállításokat.

3. lépés – A BIOS elindítja a számítógépet a legelső háttértárról, amin talál érvényes boot sector-t.

4. lépés – Lefut a boot sectorban található kód és az indítási folyamat vezérlését átveszi a Windows Boot Manager.

5. lépés – A WinLoad a BOOTMGR által megadott útvonalat használja az indítási partíció megtalálására.

6. lépés – A WinLoad betölt két fájlt, amelyek a Windows magját képezik, ezek az NTOSKRNL.EXE és a HAL.DLL fájlok.

7. lépés – A WinLoad beolvassa a rendszerleíró fájlokat (registry files), és betölti az eszköz illesztőprogramjait (driver).

8. lépés – A Windows kernelje átveszi az indítási folyamat felett az irányítást.

9. lépés – Az NTOSKRNL.EXE elindítja a WINLOGON.EXE bejelentkezésre szolgáló programot.

10. lépés – A WINLOGON.EXE megjeleníti az üdvözlőképernyőt.

**BIOS (Basic Input Output System)** – A számítógép szoftveres és hardveres része közötti interfészként szolgál.

**MBR (Master Boot Record)** - A merevlemez legelső szektorának (azaz az első lemezfelület első sávjának első szektorának) elnevezése. Ennek segítségével indul el egy operációs rendszer bootolása a számítógép bekapcsolása után.

* 1. Írja le a személyi számítógép adatvédelmi beállításait (hozzáférési jogok, fiókok, eszközök)

**Hozzáférési jogok:**

Az állományok manipulációja elleni védelem. Hozzáférési jogokat beállíthatunk adott felhasználókra, csoportokra, vagy minden egyéb felhasználóra. A hozzáférési jogok beállítása Windows és Linux operációs rendszeren eltérő.

**Windows esetén –** A következő felhasználócsoportokra tudunk hozzáférési jogokat beállítani: administrator, administrators, user, group. Ezekre a következő engedélyeket tudjuk beállítani: engedélyezés, megtagadás, olvasás, írás, módosítás, teljes hozzáférés. Ezeknek a hozzáférési engedélyeknek a tárolása az NTFS fájlrendszer ACL (Access Control List)-ével történik.

**Linux esetén –** A következő felhasználócsoportokat tartalmazza: root (gyökér), user (felhasználó), other (többiek). A következő engedélyeket tudjuk kiosztani: read (olvasás), write (írás), execute (végrehajtás), speciális engedélyek.

**Fiókok:**

**Windows esetén –** administrator, administrators, user, group

**Linux esetén –** root, user, other

**Eszközök:**

**Windows esetén –** Érdemes megemlíteni aBitLockert, amivel háttértárakat tudunk titkosítani, az AppLockert, amelynek segítségével például az alkalmazások elindítását tudjuk korlátozni és az User Account Control (UAC)-t, amivel a felhasználók jogosultságait tudjuk részletesen szabályozni (például melyik felhasználó férjen hozzá a nyomtatóhoz, stb.).

**Linux esetén –** A felhasználói jogosultságokat a „chmod” paranccsal tudjuk megadni. A jogosultságokat megadhatjuk „rwx” alakban, vagy számalakban (például 777) is.

1. Készítsen PC konfigurációt, javaslatát indokolja meg a kompatibilitás szempontjából is.

**Választott komponensek:**

**Alaplap** – MSI Z390-A PRO (Foglalat: LGA-1151)

**Processzor** – INTEL Core i9-9900K 3.60GHz (A processzor foglalata megegyezik az alaplapéval és az alaplap chipsete támogatja a processzort, tehát kompatibilisek.)

**RAM** – 2x G.SKILL 16GB Ripjaws V Red DDR4 3600MHz (Az alaplap és a kiválasztott 2 RAM memóriafoglalata DDR4. Az alaplapnak 4 memóriafoglalata van, tehát ebbe is beleférünk. A CPU és az alaplap maximálisan 128 GB RAM-ot támogatnak, szóval a 32-vel nem lesz gond.)

**Grafikus kártya** – GAINWARD 2478-BLISS GeForce RTX 3060 12GB GDDR6 Ghost OC PCIE (A grafikus kártya az alaplaphoz PCI-E csatoló segítségével csatlakozik.)

**Háttértárak**

1. SSD - SAMSUNG 250GB 970 EVO Plus M.2 PCIe M.2 (Az SSD az alaplaphoz az M.2 csatoló segítségével csatlakozik.)

2. HDD - SEAGATE BARRACUDA 1TB 3.5" 7200RPM 64MB SATA (A HDD az alaplaphoz SATA kábel segítségével csatlakozik.)

**Gépház** – COOLERMASTER Masterbox MB520 RGB (A gépház ATX formátumú alaplapokra lett tervezve, amilyen a mi általunk kiválasztott is.)

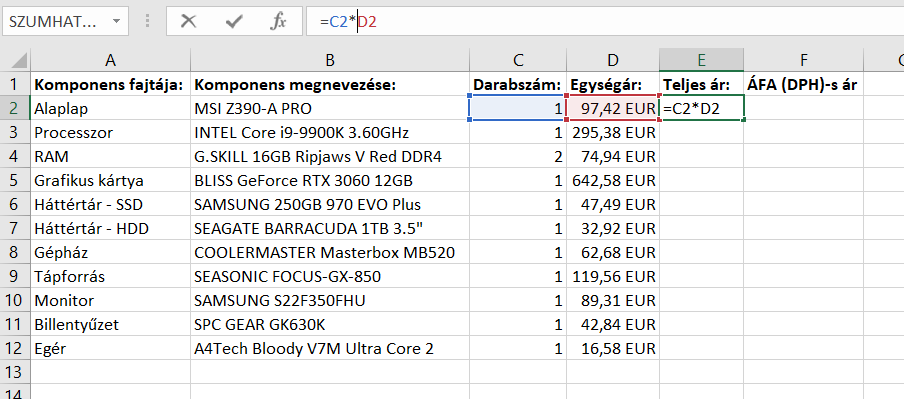
**Tápforrás** – SEASONIC FOCUS-GX-850 (Az egyes komponensek teljesítményigényét megnéztük a gyártóik oldalán, majd ezeket összeadva arra jutottunk, hogy egy 850W-os táp gond nélkül fogja bírni a terhelést.)

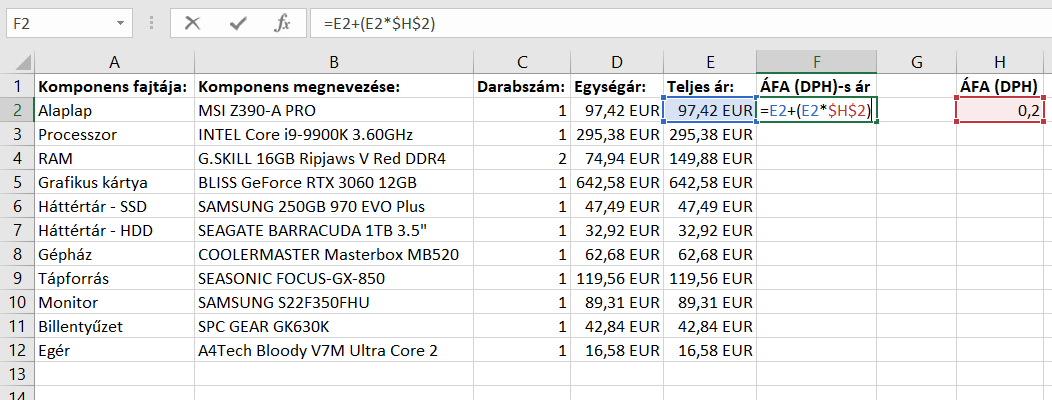
**Monitor** – SAMSUNG S22F350FHU (A monitor a videokártyához HDMI kábel segítségével csatlakozik.)

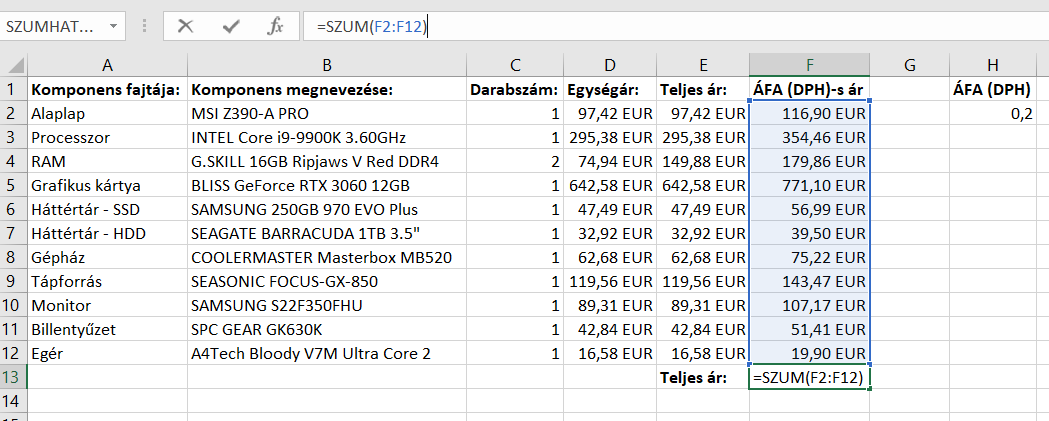
**Billentyűzet** – SPC GEAR GK630K (A billentyűzet az alaplaphoz USB kábel segítségével csatlakozik.)

**Egér** – A4Tech Bloody V7M Ultra Core 2 (Az egér az alaplaphoz USB kábel segítségével csatlakozik.)

1. Készítsen kalkulációt egy működőképes számítógép komponenseire MS Excelben.
   1. Készítsen árlistát amely tartalmazza a megnevezést, darabszámot, egységárat, teljes árat, ÁFA-s árat és a teljes összeget.

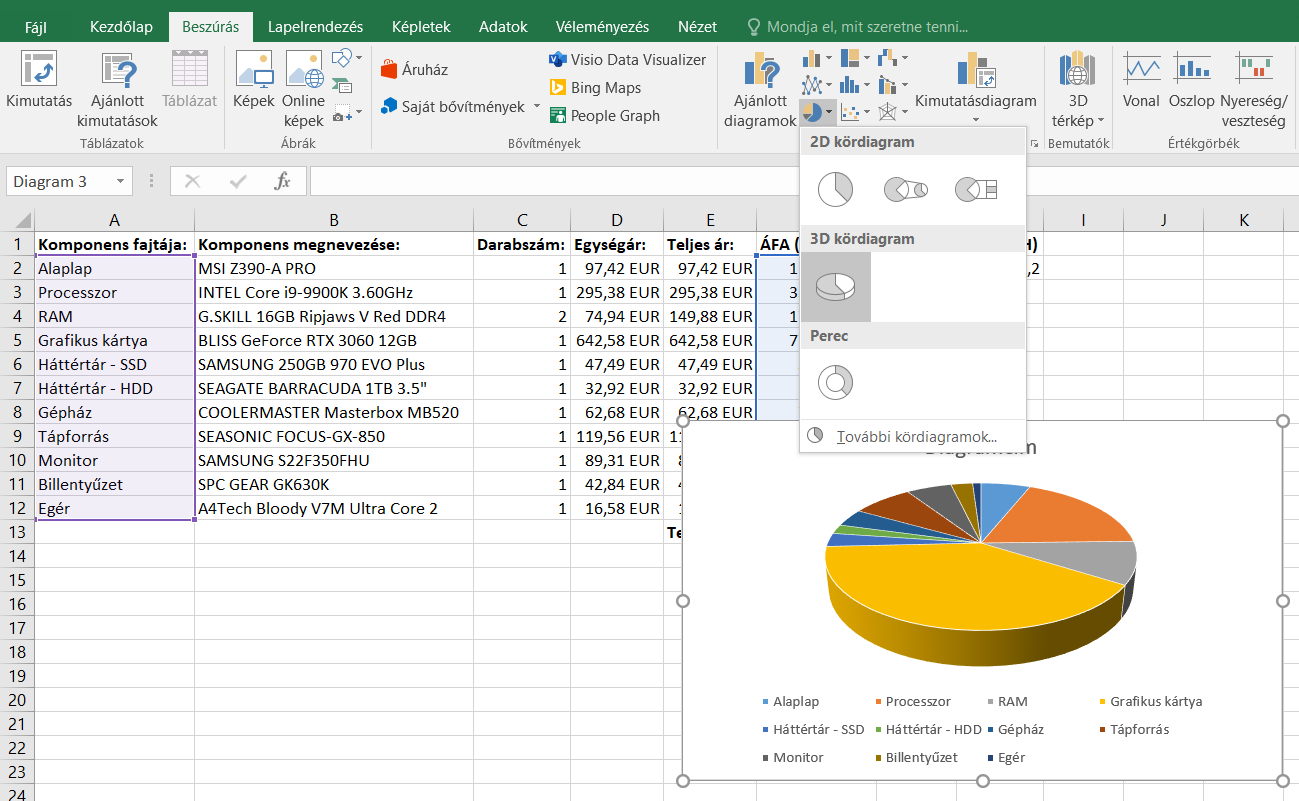
Nyitunk Excelben egy üres munkafüzetet, majd a munkalapon felvesszük az oszlopcímeket. Ezután felvesszük a komponensek neveit, melléjük pedig az árakat. Az árakat tartalmazó oszlopok cellaformátumát átállítjuk pénznemre (Ctrl+1, ezután a Pénznem opció, ezen belül pedig az Euró 2 tizedesjeggyel). A darabszámból és az egységárból kiszámítjuk a teljes (ÁFA/DPH mentes) árat a két oszlop összeszorzásával, majd dupla kattintás után az Excel villámkitöltő funkciója megismétli ezt a műveletet a többi áru esetén is.

Az ÁFA/DPH kiszámításánál hasonlóan járunk el, felveszünk egy új mezőt, amibe beírjuk az értékét (20% - 0,2), majd ezzel az értékkel beszorozzuk soronként a teljes árakat. Fontos ilyenkor az ÁFA/DPH értékét lerögzíteni F4 segítségével.

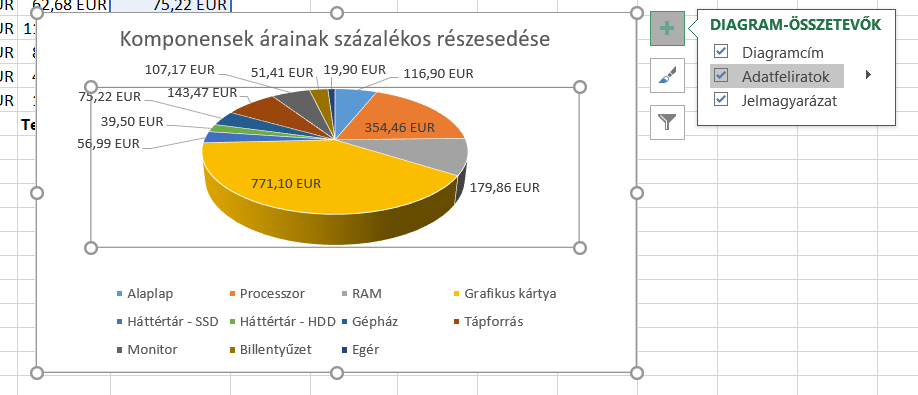
Végül utolsó lépésként kiszámoljuk a teljes árat a SZUM (angol és szlovák Excelek esetében SUM) függvény segítségével.

* 1. Kalácsdiagram segítségével ábrázolja az egyes komponensek árának százalékos részesedését.

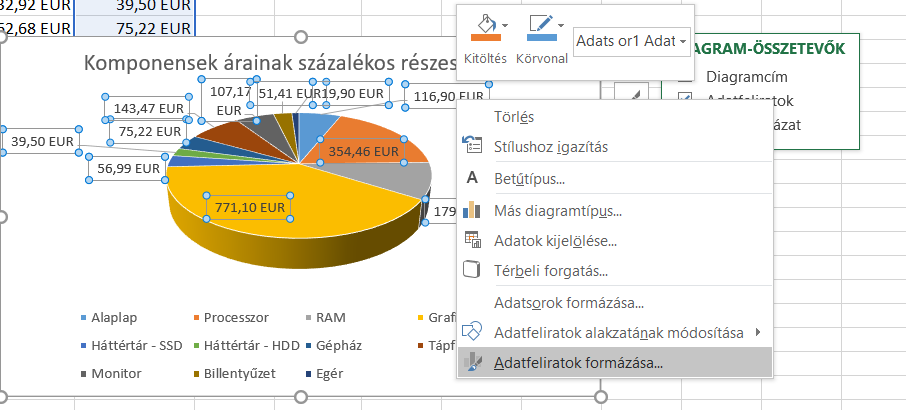
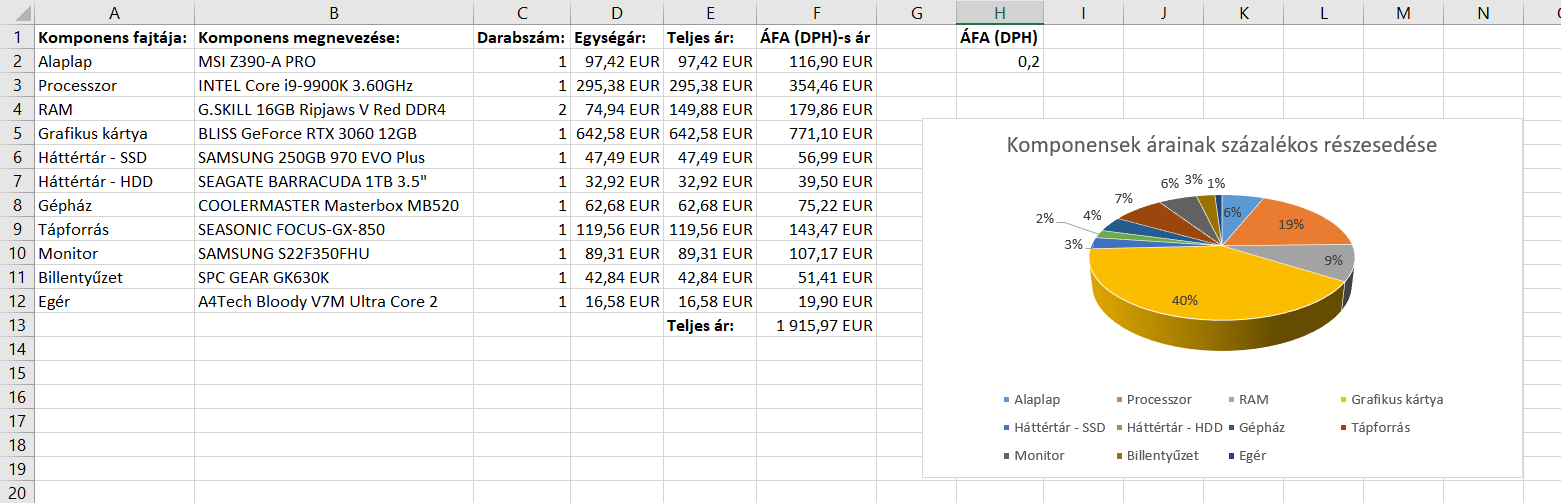
Kijelöljük a komponens fajtája és az ÁFA/DPH-s áras oszlopot, rámegyünk a beszúrásra, a Diagramok menüpontnál a kördiagramok közül kiválasztjuk a 3D kördiagramot.

*Megjegyzés: A magyar Excel nem használja a kalácsdiagram kifejezést, helyette a 3D kördiagram jelenik meg a lehetőségek között.*

Következő lépésként pedig átírjuk a diagram címét. Már csak a százalékok megjelenítése maradt hátra. Ehhez először hozzáadjuk az adatfeliratokat.



Miután ezzel megvagyunk, rákattintunk az egyik árfeliratra, lenyomjuk a jobb egérgombot, majd a legördülő menüből kiválasztjuk az utolsó lehetőséget.

A jobb oldalt megjelenő panelen először rámegyünk a százalék megjelenítésére, majd pedig az értékek megjelenítését kikapcsoljuk. Ezzel kész is lett a diagram.

1. Mutassa be a C++ programozó nyelvet (fejlődés, forráskód, a forráskód struktúrája, a forráskód fordítása)

**Fejlődés:**

A C++ egy magas szintű programozási nyelv, a C nyelv továbbfejlesztése. Az AT&T Bell vállalat fejlesztette ki a 70-es évek végén. Megalkotásánál a következő szempontokat vették figyelembe: legyen visszafelé kompatibilis és támogassa az objektumorientált programozást. Bár a magasszintű programozási nyelvekhez soroljuk, támogat számos alacsony szintű műveletet is (pl. maszkolás). Fontos eleme még az úgynevezett szabványos sablonkönyvtár (Standard Template Library – STL). Ez a könyvtár tartalmazza például a list, map, vector, stack, queue elemeket.

**Forráskód:**

A forráskód (source code) szöveges fájlként van eltárolva, ami a .cpp kiterjesztést használja. Azokat az utasításokat tartalmazza, amit a programunk segítségével szeretnénk végrehajtani.

**A forráskód struktúrája:**

**#include <iostream>**

A C++ a be- és kimeneteket adatfolyamokkal (stream) valósítja meg. Ezek az iostream előre megírt függvénykönyvtárban találhatóak. Ahhoz, hogy ezeket használhassuk, be kell illeszteni a programba – ezt nevezzük include-nak.

**using namespace std;**

A C++-ban rengeteg előre megírt funkció létezik. Ahhoz, hogy ez a sok név kezelhető legyen ún. névterekre (namespace) helyezték őket. Ahhoz, hogy az utasításoknál ne kelljen minden esetben megadni, hogy pontosan hol találhatóak globálisan az egész programunkra nézve, megadhatjuk az std névtér használatát a using kulcsszóval.

**int main()**

A main metódus a program ún. belépési pontja, ez azt jelenti, hogy amikor a programunkat elindítjuk, több különálló épp fájlból is állhat. Ennél a pontnál kezdi meg a működését. A main metódus előtt a visszatérési értéknek a típusa szerepel.

**{ - kezdő kapcsos zárójel**

Egy blokk nyitását jelenti C++-ban. A blokkokat (utasítások sorozatát) {} jelek közé tesszük. Fontos, hogy a jelekhez tartozzon nyitó és záró pár is. Fontos, ha blokkot nyitunk, a sorokat beljebb kell kezdeni.

**cout << „Hello world” << endl;**

A cout egy alapkimenetet kezelő utasítás, mely szöveges kimenet esetén az ún. konzolba (console) írja ki a kimenetet (ezért volt szükség include-ra). A kiíratás során megadjuk először, hogy kimeneti adatfolyamot szeretnénk használni. Utána átirányítjuk azokat a dolgokat, amelyeket a kimenetben szeretnénk megjeleníteni. Átirányítás jele: << - kimenet esetén a kimenet irányába mutatnak. Szöveges adatokat minden esetben idézőjelek közé tesszük. Kiíratást végző utasítás végét „”-vel látjuk el.

**return 0;**

A main függvény visszatérési értéke.

**}**

Blokk vége.

**A forráskód fordítása:**

*Megjegyzés: Lényegében erről szól a 8-as tétel, nem értem, miért tették 2 helyre is.*

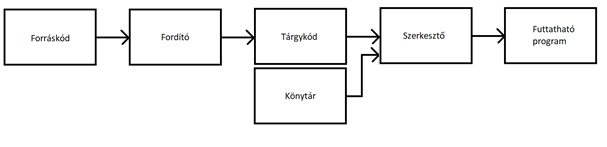
1. Mutassa be egy egyszerű forráskód kompilációját és az applikáció futtatását. Mutassa be a képernyőre való kilistázást és a billentyűzetről való adatbevitelt.

**Forráskód fordítása és az applikáció futása:**

A C++ program egy, vagy több .cpp kiterjesztésű állományból áll. Pontosan az egyik forrásfájlnak tartalmaznia kell main nevű függvényt (ez a program belépési kódja, innen indul).

Ha magasabb szintű programozási nyelvben programozunk, akkor a programozó egy ún. forráskódot ír meg. Ez a forráskód a Windows operációs rendszerben egy .cpp forrásfájlban (UNIX rendszerben: .cc) található.

A fordítás egy speciális program futtatásával történik, amit compilernek vagy fordítónak neveznek s magát a folyamatot fordításnak (kompilálásnak) nevezik.

Fordítással nem jön létre azonnal program, ennek létrehozása az ún. szerkesztő (linker) feladata. A szerkesztő feladata többek között a tárgykódhoz való más szükséges fájlok hozzácsatolása, könyvtárak, kezdeti inicializálási kód…

A **C++ forráskód** a fordítani kívánt program.

A **fordító (compiler)** az a program, amelyik a C++ forráskód fordítását végzi.

A **tárgykód (object code)** gépi utasításokat és a program memóriába olvasását meghatározó információkat tartalmazza.

A **szerkesztő (linker)** összeállítja a tárgykódot és a könyvtárakban szereplő kódokat egyetlen futtatható állománnyá (executable file). A könyvtárak előre lefordított programrészeket tartalmaznak.

**Képernyőre való kiíratás:** A cout << "Kiíratni kívánt szöveg"; paranccsal történik.

**Billentyűzetről történő adatbevitel:** A cin >> paranccsal történik, az így kapott adatot célszerű elmenteni valamilyen változóba, például stringbe.

1. Jellemezze a gazdasági rendszereket és a piaci alanyokat.

**Gazdasági rendszerek** – A társadalmak gazdasági rendszerekbe szerveződnek az alapján, hogy hányféle módon lehet megválaszolni a mit?, hogyan?, és kinek? kérdését.

**Piaci alanyok** – A piaci alanyok a háztartások, vállalatok és az állam.

* 1. Magyarázza el a hagyományokon alapuló gazdaság, tervgazdaság, piacgazdaság és vegyesgazdaság működését.

**Hagyományokon alapuló gazdaság –** Szokás és a hagyomány a meghatározó. Az idősebbektől a fiataloknak átadott hagyományok alapján annyit és úgy termelnek az emberek, mint ahogy azt őseik is tették.

**Tervgazdaság –** Gazdasági folyamatok megszervezése központilag, legtöbbször bürokratikus módon az állami intézmények által. Jellemzői az állami tulajdon, az állam tölti be a gazdasági koordinátor szerepét.

**Piacgazdaság –** A piacgazdaságban a termelők és fogyasztók összekapcsolásának fő színtere a piac. A fő cél itt a nyereségszerzés. Jellemzői a magántulajdon, a termelők közötti verseny.

**Vegyesgazdaság –** A vegyes gazdaság olyan gazdaság, amelyben vegyes koordináció valósul meg. Itt az állami és magánszféra is részt vesz a termelésben. A piaci verseny dominál, azonban kormányzati közreműködéssel. A vállalatok javarészt tehát magánkézben vannak.

* 1. Jellemezze a háztartásokat, vállalatokat és az államot.

**Háztartások –** A háztartás azoknak a személyeknek az összessége, akik függetlenül bármilyen rokoni viszonyoktól (kapcsolatoktól), egy jövedelmi, illetve egy fogyasztói közösséget alkotnak és az életviteli költségeiket folyamatosan részben vagy egészben közösen viselik. A háztartás fogalma tehát nem azonos a család fogalmával, mivel előbbi inkább gazdasági kötődés, utóbbi inkább jogi.

**Vállalatok –** Vállalatnak nevezzük a törvények alapján megalapított, jogi személyként létező társas vállalkozást. Egyének munkahelye, munkavégzés helyszíne is. Gazdasági erőt képvisel számos külső érdekelttel tart fenn kapcsolatokat. A vállalat alapítói és/vagy tulajdonosai lehetnek magánszemélyek, más vállalkozások vagy maga az állam.

**Állam -** Emberek meghatározott földterületen élő csoportjának közös kormányzattal valamint belső és külső szuverenitással rendelkező közössége. Az állam, kialakulásától fogva, nagy hatással van az uralma alatt álló területek gazdaságára. A modern állam gazdasági tevékenysége magában foglalja összes intézményének és regionális szervezetének gazdasági jellegű tevékenységét.