

1. Készítsen C programot, amely egy számológépet valósít meg a 4×4-es billentyűzet és a virtuális LCD segítségével! A számológép két egyjegyű számmal végezze valamennyi műveletét, és az ABCD→+-\* / megfeleltetésű gombok jelentsék a műveletet. Az eredmény a # gomb lenyomására jelenjen meg a virtuális LCD-n (pl. 8+4=12)!
2. Készítsen C programot, amely a memóriában eltárolt 4db 4 bites kódszót Timerrel vezérelve a P5 LED-eken kiírja. Az ain0.2 potméterével a timer sebességét állítani lehessen!
3. Készítsen C programot, amely a virtuális LCD-n folyamatosan a P4-es port értékét írja ki binárisan, hexadecimálisan és decimálisan! Pl.: 1001 1101=0x9D=157.
4. Készítsen C programot, amely a virtuális LCD-re olyan „labirintust” rajzol, ahol a „|” karakterek a falat jelölik! A bal oldalon megjelenő „O” betűt a P5 gombjaival lehessen mozgatni (P5.0 jobb, P5.1 le, P5.2 fel), a falakra nem lehet lépni. Ha az „O” elérte a „C” célt, írja ki, hogy vége a játéknak! Példa:

O															
															C

5. Modellezzen az LCD kijelzőn egy FIFO adatátviteli csatornát! A kijelző 1. szegmensébe mint pufferbe **kerüljön** a Keypadon beadott karakter, ami 1 s időzítéssel (Timer0) **lépjen tovább** a következő szegmensbe, majd a 32. szegmens után lépjen ki a csatornából! A kijelző 1. szegmense folyamatosan várja a bemenő adatot, azonban ha a Keypadról nem érkezik input, akkor üres karakter (szóköz) kerül továbbításra a csatornán keresztül.
6. Készítsen beléptető programot C nyelven a 4×4-es billentyűzet és a virtuális kijelző segítségével! A belépéshez a program **kérjen be egy négy számjegyű kódot**, majd a # gomb lenyomásával **nyugtázza** a bevittet. A kód 4 karaktere \*-ként jelenjen meg minden leütés után az LCD felső sorában! Amennyiben a helyes kódot ütöttük be, a # lenyomása után az LCD alsó sorába **írja ki, hogy „ACCEPTED!”**, ellenkező esetben (illetve ha nem a # gombot ütöttük be 5.-ként), az LCD alsó sorába **írja ki, hogy „DENIED!”**. A felirat **3 másodpercig legyen kint** a kijelzőn, utána álljon vissza a program a kiinduló állapotba!
7. Készítsen C programot, amely a P4-es port kapcsolóinak **aktuális 0/1 állapotait** (OFF/ON állapot) jeleníti meg a virtuális LCD kijelző felső sorának első 8 szegmensén! A megjelenítést a **potenciométer tekerésével szabályozza** úgy, mindig csak **egyetlen kapcsoló állapota** legyen látható az LCD-n, és **bal végállásban a bal szélső, jobb végállásban pedig a jobb szélső** kapcsoló állapota legyen látható! A többi kapcsoló állapotát ne mutassa az LCD!  
Pl.: **1 0 0 1 1 1 0 0**
8. Készítsen C programot, amely a potenciométer segítségével egy V43-as sorozatú mozdony controllerét szimulálja, és jelzi a virtuális kijelzőn a következőképpen:
  - Ha a P4.4 kapcsoló ON, akkor a mozdony **ELŐRE** irányban halad, a kijelzőn az ELORE és a FOKOZAT: **érték** (0%-100%) jelenik meg –tetszőleges helyen.
  - Ha a P4.4 kapcsoló OFF, akkor a mozdony **HÁTRA** irányban halad, a kijelzőn a HATRA és a FOKOZAT: **érték** (0%-100%) jelenik meg –tetszőleges helyen.
  - A P4.4 kapcsoló átkapcsolása csak akkor lehetséges, ha a FOKZAT érték 0%. (Vagyis menet közbeni átkapcsolásnál az irány nem változik meg, az csak akkor fog, ha letekerjük a kontrollert, és a következő feltekerésnél vesszük csak figyelembe.)

- A jármű álló helyzetből indul, azaz bekapcsolás után csak akkor van irány, ha a menetkapcsolón a fokozat 0!
- 3 másodpercenként a NYUGTA felirat jelenjen meg a kijelzőn, amelyet 3 másodperc alatt nyugtázni kell a P5.0 gomb lenyomásával. Ha nem történik nyugtázás, az irány eltűnik (vérszéfékezés) – éberségi funkció. Vérszéfékezés után a jármű újraindul, ha a P5.0-ás gombot megnyomtuk!



9. Készítsen C programot, amely egy jármű kijelzőjét szimulálja a virtuális kijelzőn a következőképpen:

- A program elején válasszuk ki a járatszámot és a végállomást (tetszőleges, de férjen ki az első sorba, azaz max. 16 karakter!); Ehhez definiáljunk előre 3 lehetséges változatot, amelyeket sorban 3 másodpercenként – egymás után - kiírunk a kijelző felső sorába, és ha valamelyik esetén lenyomjuk a P5.3-at, akkor az rögzül az első sorban!
- A rögzítés után a jármű elindul, és az útvonalán 4 megállója van (ebből a 4. a végállomás), mindegyiknek van neve (max. 16 karakter) és átszállási lehetőségei (max. 16 karakter). A megálló nevét és a hozzá tartozó átszállási lehetőségeket 4 másodperces váltásban írja ki a kijelző második sorára. (Rögzítés után automatikusan az első megálló és átszállási kapcsolatai kerülnek kijelzésre.)
- A következő megállóra ugrás a KeyPad A gombjának lenyomásával (ha van ilyen), míg az előző megállóra ugrás a KeyPad B gombjának lenyomásával (ha van előző megálló) történjen!
- Egy gombnyomásra csak egyet lépjen tovább a program!
- A KeyPad D gombjának lenyomása után kezdődjön előlről a program, azaz kérje a viszonylat kiválasztását!

