

Vége a szótlan programozásnak!

Beszéd felismerő programmal egy lépéssel közelebb HAL-hoz.



Ugye, François, ez csodálatos? Kisgyermekkorom óta – még mielőtt ennek az étteremnek a megnyitására gondolhattam volna – egyfolytában valami ehhez foghatóról álmodoztam. Ahogy visszaemlékszem, a 2001: Űrodüsszeiát néztem, a HAL 9000-es számítógép hangját hallottam, és arra gondoltam: „Ez az, amit szeretnék! Egy beszélő számítógép!” Évek múlva rá kellett ébrednem, hogy a számítógépem még mindig nem képes beszélni. Elérkezett azonban a nap, amikor mindez megváltozhat. Hogyan? Már itt is vannak a vendégeink? Isten hozott benneteket Chez Marcelnél! Nagyon

örülök az érkezéseiteknek! Ma az igényes programozó csodálatos lehetőségek közül választhat az étlapon. Foglaljatok helyet, François rögtön hoz valamilyen finom bort. François, hozz fel a pincéből egy palack 1996-os évjáratú Ausztráliából származó Hill of Grace-t! Kedvelni fogjátok ezt a pompás ízt. Köszönöm, François. Igen, tölts kérlek a vendégeinknek!

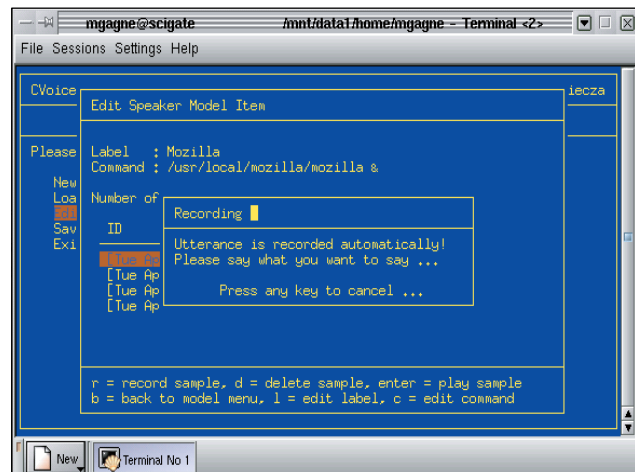
Épp az imént mondtam Françoisnak, hogy manapság már mindenféle beszélő számítógépeket kellene használnunk, de a linuxos gépem mégiscsak szótlannal dolgozik. Számomra érthetetlen, miért nem képes a beszédre egyetlenegy rendszeremre telepített program sem! Nos, mai receptjeinkhez a számítógépedbe szerelt és megfelelően beállított hangkártyára, valamint egy mikrofonra lesz szükséged. A skóciai Edinburghi Egyetem Beszédtechnológiai Kutató Központja (CSTR) pontosan azt kínálja, amivel nekivághatunk a vágyott útnak – amelynek megtétele után már a saját számítógépünk is beszélni lesz képes! Aki ellátogat a <http://www.cstr.ed.ac.uk/projects/festival> címre, rábukkanhat a Festival elnevezésű izgalmas vállalkozásra. A Festival többnyelvű beszédkészítő rendszer, amely számos emberi nyelven képes a begépett szöveget beszéddé alakítani. A program API-felületével több további programba és alkalmazásba illeszthető be. Amint ezt a csomagot kibontjuk, rögvest megláthatod, mit is értek ez alatt. A Festivalhoz könnyű hozzájutni, csak meg kell keresned a fent említett helyet a Világhón, majd kattints a letöltés gombra – és máris a programcsomag legfrissebb változatát kapod kézhez. Ha már úgyis ott jársz, a `speech_tools` eszközkészletet is vedd magadhoz, mivel a Festival feltételezi ennek meglétét. Az első program, amelyet összeépítünk, következképpen az alábbi kell legyen:

```
tar -xzf speech_tools-1.2.1.tar.gz
cd speech_tools
cp config/config-dist config/config
chmod +w config/config
```

Ezen a ponton eldöntheted, szeretnél-e megosztott könyvtárakat használni, az alapértelmezés ugyanis ezt nem teszi lehetővé. Amennyiben igényled, a `#` jel törlésével a `config/config` állományban meg kell szüntetni a megjegyzést.

```
#SHARED=1.
```

Ez javasolt választás, és akár élsz ezzel a lehetőséggel, akár nem, az összeépítést tovább folytathatod.



Parancs hozzáadása a `model_editor`-ban

```
make info
make
make install
```

Elérkezett a lazítás ideje: kóstoljuk meg a libamját (ez csakugyan finom, nem igaz?) és kortyintsunk egyet a borunkból! Amint a `speech_tools` összeépítése befejeződött, belekezdhetünk a Festival-rendszer létrehozásába. Bontsuk ki a rendszer forráskódját egy tetszőlegesen kiválasztott könyvtárba

```
a tar -xzf festival-1.4.1.tar.gz parancsral.
```

E műveletet követően a benne szereplő összes állomány a `/festival` könyvtárba kerül. Mielőtt bármi mást tennénk, végezzük el a nyelvi lexikon és a beszédadatbázis kibontását is. A munkát az alábbi állományokkal kezdtem el:

```
festlex_CMU.tar.gz
festlex_POSLEX.tar.gz
festvox_kallpc16k.tar.gz
```

A CMU elnevezésű könyvtár az angol beszédhangok, a POSLEX könyvtár pedig az általánosan használt angol nyelvi elemek tárolására szolgál. Végül helyére kerülhet a 16 k-s mintavételezéssel készült amerikai angol kiejtés minta-adatbázis is. A világ más-más részén élő olvasók kétségkívül eltérő igényekkel lépnek majd fel a beszélők hangját, nemét vagy nemzetiségét (angol vagy francia) illetően. A szükséges feltételek részleteiről a Világháló fent említett helyén, a `Distribution` könyvtárban található README állományból tudhatunk meg a legtöbbet.

Ha ezeket az állományokat kibontod, a művelet befejezése után ugyanebbe a `/festival` könyvtárba fognak kerülni. Innentől kezdve a folyamat megegyezik a `speech_tools`sal végzett műveletsorozattal, a `config-dist` könyvtárból való `config` állomány másolásával bezárólag, amit természetesen ezúttal is a `/festival` könyvtárba másolunk. Miután mindezzel végeztünk, a telepítő könyvtárból adjuk ki a `bin/festival` parancsot. Ekkor várhatóan a következő sorok fognak megjelenni a képernyőn:

```
bin/festival
Festival Speech Synthesis System 1.4.1:
release November 1999
Copyright (C) University of Edinburgh, 1996-1999.
All rights reserved.
For details type `(festival_warranty)`
festival>
```

Felkészültél arra, hogy most valóban *hallani* fogod, hogyan beszél a számítógép? Ha a kérdésre igen a válasz, akkor a parancssorjelnél (>) gépeled be a következőket (ügyelj a zárójelekre és idézőjelekre, mert ezek is fontosak!):

```
(SayText "François. Vite. More wine.")
```

Feltéve, hogy minden megfelelően zajlott, a hangszórókból a következő szavakat hallod: François. Vite. More wine. Úgy találd, hogy meglehetősen parancsoló ez a hangnem? Részemről szólva szeretek ezzel a sorral játszani... hűségesebb pincéremet kissé kihozza a sodrából. Angol kiejtést és adatbázist használok, bizonyára ennek köszönhető, hogy a hanglejtés és a hangsúlyozás érdekesen hat. A CONTROL+D billentyűk együttes lenyomásával lehet kilépni a Festival parancsmódú végrehajtásából, de a *quit* is használható erre. Szeretném még egyszer a zárójelek fontosságára felhívni a figyelmet. Most próbáljunk ki valami érdekeset! A `--tts` kapcsoló segítségével megadhatjuk egy szövegállomány elérési útját és rábírhadjuk a Festivalt, hogy a kiválasztott szöveget felolvassa nekünk. Vegyük például a `fortune` program lefuttatásával minden éjjel frissíti a nap üzenetét. Az üzenetet az alábbi módon olvastathatjuk fel:

```
bin/festival --tts /etc/motd
```

Ha a parancs begépelésekor nem adunk meg állománynevet, máris elkezdhetjük a beírást, a szöveg végére érve pedig nyomjuk le a CONTROL+D billentyűkombinációt – ezt követően a Festival program végrehajtása befejeződik. Akad itt még valami, amit érdemes kipróbálni: vezessük csövezetékbe a Festival program kimenetét! Kíváncsi vagy esetleg a dátum valamilyen egyéni hangú értelmezésére? Ha igen a válaszod, próbáld ki: `date | bin/festival --tts`. A Fortune programot akkor is érdemes futtatni, ha ennél azért nagyobb bölcsesség birtokába szeretnél jutni:

```
/usr/games/fortune | bin/festival --tts
```

A Festival a `--server` kapcsoló megadásával kiszolgálóként is működtethető, így további program(ok) számára képes szöveges adatokat közvetíteni. Készíthető például olyan program, amely kimenetét a Festival csatolójára küldi: ez alapértelmezés szerint az 1314-es kapu. A *lista* egy apró Perl-programot tartalmaz, amelyet pontosan e célból írtam. Nem különösebben érdekes, de tekintheted alkalmazásaid kiindulópontjának. Ne feledd meg arról, hogy első alkalommal szükséges lehet a Perl végrehajtható modulok elérési útjának megváltoztatása.

Ezt az ötletet bámulatra méltó módon valósítja meg a Darxus által írt *speechd* nevű program. Ez olyan készülék-leíró állományt telepít – a `/dev/speech` névvel hivatkozhatunk rá –, ami tetszőleges szöveg fogadására képes. Irányítsuk át a programkimenetet erre a készülékre, ezt a kiszolgálómódban futó Festival program kapja meg, majd hangosan felolvassa. Ez megint csak egy Perl-héjprogram, amit a SpeechIO szervezet honlapjáról tölthetsz le, címe <http://www.SpeechIO.org/>.

A tömörített állományokat csomagoljuk ki egy ideiglenes könyvtárba, és ott végezzük el a program összeépítését, az alábbi módon:

Írás a Festival program csatolójára

```
#!/usr/bin/perl
#
# Írás a Festival program csatolójára
# Marcel Gagne - 2001
#
$remote_port="1314";
$festival_host="localhost";
$my_message="Francois, come here. I need you.";

use IO::Socket;
$socket = IO::Socket::INET->new
    (PeerAddr => $festival_host,
    PeerPort => $remote_port,
    Proto    => "tcp",
    Type     => SOCK_STREAM)
    or die "Nincs élő kapcsolat".
$remote_host:$remote_port : $@\n";

# Írás a Festival kiszolgáló kapura
if ($socket) { # Csak ellenőrzés
    print($socket `(SayText '$my_message')`
or
        die "Fut a Festival kiszolgáló?\n";
}

# Csatlakozás lezárása, ha még nem volt
# lezárva.
close($socket);
```

```
tar -xzvf speechd-0.54.tar.gz
cd speechd
make
make install
```

A program démonként való futtatásához a parancssorba a következőt gépeljük be:

```
/usr/local/bin/speechd.
```

Mielőtt folytatnám, el kell mondanom, hogy itt gondjaim támadtak. RedHat-rendszerem `/etc/hosts` állományában az alábbi bejegyzés szerepelt:

```
127.0.0.1 localhost.localdomain localhost
```

Elképzelhető, hogy a te rendszerednél is hasonló a helyzet. A helyes működéshez a két meghatározást a következő módon fel kellett cserélnem:

```
127.0.0.1 localhost localhost.localdomain
```

Előfordulhat, hogy a Festival kiszolgálóprogram éppen nem működik, ekkor ezt a *speechd* megpróbálja majd elindítani. Abban az esetben, ha a Festivalt forráskódból építetted fel, sajnos szükséges lehet a *speechd* héjprogramjának olyan módosítására, hogy a végrehajtható programmodul teljes elérési útját használja. A másik lehetőség a Festival bináris moduljainak a `/usr/local/bin` könyvtárba történő másolása. De mit lehet ezzel kezdeni? Visszatérve a Fortune-progra-

mos példához, a program kimenetét egyszerűen átirányíthatom a /dev/speech állományba:

```
/usr/games/fortune > /dev/speech
```

Ezt már roppant könnyű lesz beépíteni a héjprogramjaidba. Álljon itt egy további példa is! Lehetőség nyílik rá, hogy egy héjprogramon – amely néhány percnként lefut és ellenőrzi a bejövő postát – megszámlolja az üzeneteket, továbbá hogy ezt a beszédkészüléken keresztül hangosan fel is olvassa nekünk. Fontos megjegyezni, hogy az `frm` parancs tulajdonképpen fordított percejek.

```
echo "You have `frm | wc -l` messages in your
↳ mailbox" > /dev/speech
```

Miután Linux-rendszerünket sikerült szóra bírni, úgy tűnik, már csak egyvalami hiányzik. Meg kell tanítanunk a rendszert, hogy „megértse” az emberi szót és végrehajtsa, amit mondunk neki, vagyis *beszédfelismerő* programra van szükségünk. A szóban forgó testreszabás megvalósításához a Világhálón felkerestem *Daniel Kiecza* honlapját a beszédalapú irányítóprogram, a *cvoicecontrol* forráskódjáért, ezt a csomagot ugyanis a GPL-szabályok szerint terjesztik. A *cvoicecontrol* segítségével belefoghatunk álmaink rendszerének létrehozásába. Természetesen ezentúl már igencsak óvatosnak kell lennünk, hogy mit mondunk a számítógép közelében!

```
tar -xzvf cvoicecontrol-0.9alpha.tar.gz
cd cvoicecontrol-0.9alpha
./configure
make
make install
```

A parancsok végrehajtása nyomán keletkező állományok a /usr/local/bin könyvtárban jelennek meg. Három tényezőt célszerű figyelembe venni: az első mindjárt maga a *cvoicecontrol* program. Használat előtt állítsuk be a mikrofont és hozzunk létre hangminta-állományokat! E feladatokat a *microphone_config*, illetve a *model_editor* programokkal leszünk képesek megoldani. Kezdjük a beállítást a *microphone_config* program segítségével, a párbeszéd végigkövetésével. Az alapértelmezett audiokészülékeket és keverőberendezéseket a rendszer önműködően érzékeli (elvileg már a megfelelő helyen ki vannak töltve). Az én rendszeremben ezek */dev/mixer*-ként és */dev/dsp*-ként jelentek meg. A következő lépés a keverőberendezés szintjeinek beállítása, a hangfelvételek zajszintjének meghatározása és a beállítóállomány létrehozása. Ebben a szakaszban valószínűleg az a legnehezebb feladat, hogy eléggé hangosan legyünk képesek beszélni mindaddig, amíg a program beállítja a működési jellemzőket. Már most elárulhatom, hogy ez sokkal nehezebb feladatnak fog bizonyulni, mint amilyenek hangzik. Ha az állomány elhelyezése felől külön nem rendelkezünk, akkor az állománynak az elérési útvonala a \$HOME/.cvoicecontrol/config lesz. Ezután kezdődjék a móka: indítsd el a *model_editor* programot! A képernyőn megjelenik a menü, amelyen már létező hangmintákat tudsz betölteni vagy újakat alkothatsz, szerkeszthetsz és menthetsz el. Vess egy pillantást *képünkre*, amelyen működés közben láthatod a programot. Rajta, barátaim, most a ti hangotokat fogjuk mintaként használni! Mínthogy egyelőre semmilyen beszédmintánk sincs, válasszuk a *New Speaker* (Új beszélő) lehetőséget, ezt követően pedig az *Edit*-et (Szerkesztést). Az újabb menüben szavakat rögzíthetünk és megadhatjuk, hogy mi történjen ezek kiejtésékor. Míndezt egy billentyű, az `^` (az angol *add* szóból elvonva) lenyomásával érhetjük el, amellyel új eseményt tudunk megadni. Ez először olyan általános elemként fog megjelenni a listában, amelyhez még nem tartozik parancs. Az ENTER gomb lenyomását követően beállíthatók az ese-

mény jellemzői és beszédes címke adható, amely a megvalósított szolgáltatást jelzi. Egy ilyen indítóelemet hoztam létre például a Mozilla indításához és elneveztem *Start Mozilla*-nak, ezt követően pedig begépeltem a `/usr/local/mozilla/mozilla & A Mozilla indítása` parancsot. Lényeges, hogy a parancs végénél „és” (&) karaktert írjunk, mert így a parancs végrehajtása háttérben zajlik, valamint a program elindulása után további beszédfeldolgozó parancsokat tudunk majd kiadni. Ha eddig eljutottál, most létre kell hoznod a hangmintáidat, mégpedig legalább négyet. Ejtsd tisztán a parancsok nevét, tarts némi szünetet, majd vedd fel a hangmintát a listába! Amint mind a négy megvan, üsd le a `B` (az angol *back* szóból rövidítve) billentyűt a kilépéshez és mentsd lemezre a hangminta-készletet. Az állomány nevére semmilyen kötöttség nem vonatkozik, csak az a fontos, hogy ne felejtsük el a pontos helyét. Ezzel befejeztük a mikrofon beállítását, és egy olyan számítógépes parancsral rendelkezünk, amelyhez szóbeli utasítás kapcsolódik. A Mozilla elindításához egyébként akár a *browser* (böngésző) szót is használhatjuk, de a programokat többnyire mégiscsak a nevük szerint szokásos és érdemes kiválasztani. Végül indítsuk el a *cvoicecontrol* programot!

```
cvoicecontrol speakermodel.cvc
```

Hangmintakészletem mentésekor a nevemet követően (ezt nem szükséges kitölteni) a `.svc` kiterjesztést adtam meg, és a beszédfelismerő programot a *cvoicecontrol chefmarcel.svc* parancsral tudtam üzembe helyezni. Ezután csak annyit kell mondanom: *Mozilla* – és a böngésző program máris munkához lát. Ilyen parancsállományokat készítettem kedvenc szövegszerkesztőmhöz, a `vi`-hoz és természetesen néhány játékhoz is.

Pingvin, kérlek, tárd ki az étterem ajtaját! A mai menükön szereplő eszközökkel elindulhatsz azon az úton, amelyen járva már ma a tiéd lehet a jövő számítógépe. Hiszen Linuxot használsz – szinte máris ott vagy! Kedves barátaim! Pingvin pajtás azt mondja, későre jár és sajnálunkra rövidesen itt a záróra. De túlságosan korán nincs értelme bezárni. François, nem töltenél még egy pohárral a vendégeinknek? Hogyne tudnám, François, hogy ember vagy és nem gép! Ne nézz már olyan sértődötten! Csak vicceltem. Barátaim, még egyszer köszönöm, hogy eljöttetek. Legközelebb is találkozzunk itt, Chez Marcelnél. Kedves egészségetekre!



Marcel Gagné (mggagne@salmar.com) Mississaugaban (Ontario, Kanada) él, a Salmar Consulting Inc. cég elnöke. A cég rendszerépítéssel és hálózati tanácsadással foglalkozik. Marcel pilóta és író egy személyben (tudományos-fantasztikus regényeket ír), társszerzője egy sci-fi, fantázia- és horrorantológiának, a TransVersionsnak. Kedveli a Linuxot és a Unix minden változatát. Mostanában a *Linux System Administration: A User's Guide* című művén dolgozik. A világhálón elérhető honlapján sok hasznos dolgot találhatunk. ☞ <http://www.salmar.com/marcel/>

Kapcsolódó címek

Festival honlap a Világhálón
☞ <http://www.cstr.ed.ac.uk/projects/festival/>
SpeechIO (a *speechd* honlapja a Világhálón)
☞ <http://www.SpeechIO.org/>
A borok főhadiszállása ☞ <http://www.winehq.com/>