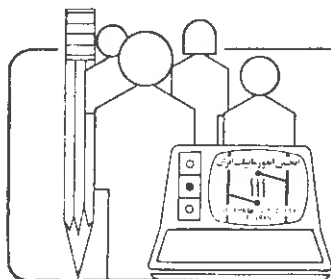


گزارش کامپیوتر

ماهنامه انجمن انفورماتیک ایران



سال هفتم - شماره ۸

شماره پیاپی ۷۰

آبان ۱۳۶۴

تکثیر ۱۰۰۰ ریال - اشتراک سالانه ۸۰۰ ریال



اخبار انجمن

تشکیل جلسه مجمع عمومی

هفتمین مجمع عمومی انجمن انفورماتیک ایران از ساعت ۱۶ تا ۱۸/۵ روز چهارشنبه ۱۵ آبان ۱۳۶۴ در محل وزارت فرهنگ و آموزش عالی تشکیل گردید. نظریه‌ای که جلسه‌ایین مجمع در دعوت اول (چهارشنبه ۱۷ مهرماه ۶۴) رسمیت نیاخته بود، جلسه پس از شروع با عده حاضر، رسمی اعلام شد.

مواردی که در دستورالعمل جلسه قرار داشتند، عبارت بودند از:

- ۱ - افزودن مواد جدیدی به اساسنامه انجمن.
- ۲ - گزارش مالی و تصویب ترازنامه.
- ۳ - از دیباچه عضویتها.
- ۴ - برگزاری انتخابات هیأت اجرایی.
- ۵ - بحث آزاد پیرامون فعالیتها و خط مشی انجمن در سال آینده.

گزارش کامل این جلسه، در همین شماره ماهنامه درج گردیده است.

بدین وسیله مسئولین وزارت فرهنگ و آموزش عالی و به ویژه آقای لاجوردی که در زمینه برگزاری جلسه مجمع عمومی انجمن و فراهم نمودن امکانات لازم همکاری نمودند، تشکر می‌نمایم.

آخرین جلسه هیأت اجرایی قبلی انجمن

آخرین جلسه هیأت اجرایی قبلی انجمن در تاریخ ۶۴/۱۵/۹ تشکیل گردید. در این جلسه در مورد چگونگی تحویل کارها به اعضای جدید هیأت اجرایی بحث و تبادل نظر گردید و مقرر شد تا سرپرستان فعلی کمیته‌های انجمن، در اولین جلسه هیأت اجرایی جدید حضور یابند و پس از تعیین سرپرستان جدید کمیته‌ها، در مورد تحویل کارها همکاری لازم را مبذول نمایند.

انتشار مجله "پژوهش کامپیوتر"

در آخرین نشست هیأت اجرایی قبلی انجمن، انتشار یک مجله پژوهشی و تحقیقاتی در کنار "گزارش کامپیوتر"، به تصویب رسید. برای این مجله که حاوی مطالب پژوهشی و تخصصی در رشته کامپیوتر خواهد بود و قلمداد "دریاد" ورق بزنید



تصویری جلد به مناسبت درج مقاله "جنبه‌های کامپیوتری طرح دفاع ضد موشکی"

دو این شماره

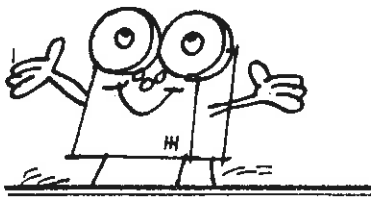
- اخبار انجمن ... تشکیل جلسه مجمع عمومی، انتشار مجله "پژوهش کامپیوتر" ۱
اخبار جهان چاپ رنگی با کیفیت و قیمت بالا، سونی و تولید دیسکهای لیزر ۱۲
میکروسکوپ پوینگر، هند در صد توسعه تکنولوژی ساخت کامپیوتر ۱۹
گزارش تشکیل جلسه مجمع عمومی تمديد عضویتها ۲۲
اعضاء تغییرات در آمار اعضای انجمن، (ترجمه دکتر بهروز پرها می) ۲۲
نکته فرصت طلبی در بار کامپیوتر (ترجمه دکتر بهروز پرها می) ۲۱
نامه ها پاسخ به سوالات علمی و فنی خوانندگان - کامی به سوی کامپیوترها ۲۴
شخصی : قسمت آخر (ترجمه محمد حسن برادران قاسمی) ۲۴
سرگرمی بازی "نیم" (گردآوری و ترجمه دکتر پروین صر) ۳
مقاله ۱ مروری بر جنبه‌های کامپیوتری طرح دفاع ضد موشکی ایالات متحده ۷
مقاله ۲ آبی بی ام به آزمایشگاههای تکنولوژی پیشرفته ژاپن راه می‌یابد (ترجمه محمد میرزا عبداللهی) ۸
مقاله ۳ زبانه‌های مبتنی بر جریان داده‌ها (ترجمه میترا آل یاسین) ۱۴
مقاله ۴ مروری بر صنعت داده پرداز ای اروپا در سال ۱۹۸۴ (ترجمه لیلی حاجی اسماعیلی) ۱۵
مقاله ۵ مفاهیم اولیه سیستمهای عامل : قسمت دوم (ترجمه عباس مصطفوی سبزواری) ۲۰
مقاله ۶ آشنائی با پیشتازان علم کامپیوتر - قسمت هفدهم (ترجمه ابراهیم نقیب زاده مشایخ) ۲۰
جان ویلیام ماکلی : سازنده اولین کامپیوتر رقمی

دنباله اخبار انجمن ...

امر، هر سه تا چهار ماه یکبار منتشر خواهد گردید، نام "پژوهش کامپیوتر" انتخاب گردیده است. چگونگی انتشار، انتخاب سردبیر، و تأمین بودجه این نشریه جدید انجمن، به هیأت اجرایی جدید، واگذار گردیده است.

قابل توجه کسانی که تمديد عضویت نکرده اند

کسانی که عضویت خود را در انجمن تمدید نکرده اند، تنها شش ماه مانده به گزارش کامپیوتر را دریافت خواهند داشت. پیش از این برای این گونه افراد تا یکسال مانده به ارسال می‌گشت. البته هنوز مطابق اساسنامه کسانی که عضویت خود را تمدید نکردند تا یکسال عضو انجمن محسوب می‌گردند.



لیست مقالات واصله

مقالات زیر تا تاریخ ۶۴/۸/۱۵ به کمیته انتشارات انجمن واصل گشته اند:

- ۱ - مقدمه ای بر نسل پنجم ترجمه دکتر محمود نقیب زاده
- ۲ - روشهای اشکال زدائی برنامه ها ترجمه ابراهیم نقیب زاده مشایخ
- ۳ - ارگونومی در کامپیوتر ترجمه کارن آقا جانی - مهران هخامنشی
- ۴ - مفاهیم و تواناییهای یک ماشین بانک اطلاعاتی ترجمه میترا آل یاسین



تا لحظه ای که ما هنا مده زیر چاپ می‌رفت (۶۴/۸/۱۹)، هنوز اعضای هیأت اجرایی جدید انجمن تشکیل جلسه نداده و مسئولین جدید و سرپرستان کمیته های مختلف انجمن را از میان خود انتخاب نکرده بودند. طبق تصویب هیأت اجرایی قبلی، این کار با پرداخت کثرت ۱۰ روز پس از برگزاری انتخابات هیأت اجرایی صورت پذیرد.

بدین جهت، اسامی رئیس، نایب رئیس، دبیر، خزانه دار، و سرپرستان جدید کمیته های انجمن، در شماره آینده ما هنا مده اطلاع اعضای گرامی انجمن خواهد رسید.



نشریات و کتابهای واصله به انجمن

- ۱ - راهنمای نوشتن رساله های پژوهشی مرکز اسناد و مدارک علمی کشور
- ۲ - راهنمای سمینارها، کنفرانس ها و سمپوزیوم های برگزار شده در ایران مرکز اسناد و مدارک علمی
- ۳ - تازه های کتابخانه، دوره ۲ شماره ۲ مرکز اسناد و مدارک علمی



باز هم تغییر در اساسنامه

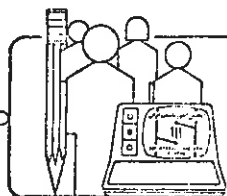
همان گونه که اعضای گرامی انجمن در گزارش مجمع عمومی ملاحظه خواهند نمود، مواد جدیدی از سوی وزارت کشور و شهر با نی جمهوری اسلامی ایران در نظر گرفته شده و مقرر گردیده است تا انجمن های علمی، این مواد را در اساسنامه های خود بگنجانند. این برای چندمین بار ظرف سالهای اخیر است که چنین دستوری از سوی وزارت کشور صادر گردید و عدم اجرای آن موجب عدم صدور پروانه برای انجمن و لاجرم، جلوگیری از ادامه فعالیتها می‌گردد. بحث بر سر محتوای مواد را نه شده و منطقی یا غیر منطقی بودن آنها نیست بلکه مسئولین محترم وزارت کشور و شهر با نی جمهوری اسلامی با بدین نکته در نظر داشته باشند که اساسنامه یک انجمن علمی، حاوی مواد نسبتاً "لاابحار" و ثابتی است که نباید هر سال و هر لحظه دچار تغییر و دگرگونی گردد. وانگهی، به دلیل آن که گنجانیدن مواد جدیدی در اساسنامه و بطور کلی هر گونه تغییری در آن، مستلزم تشکیل مجمع عمومی جهت تصویب این امر می‌باشد، با توجه به امکانات بسیار محدود انجمن های علمی، امکان برپایی و برگزاری مجمع عمومی در هر لحظه موجود نیست.

قابل ذکر است که برخی از مواد که هر بار به عنوان ماده ای جدید ارائه می‌گردند، تنها تغییر لفظ یا فتوای ترکیبی از مواد قدیمی می‌باشد، و کلیه آنها در رابطه با آن بخش از فعالیتها و انجمن های علمی هستند که به نوعی با سازمانها و ارگانهای کشوری ارتباط پیدا می‌کنند. بنا بر این، پیشنهاد ما به مسئولین وزارت کشور این است که به منظور کمک به انجمن های علمی و تسهیل در فعالیت آنها، آئین نامه ای جهت فعالیت انجمن های علمی تنظیم نمایند و این گونه مواد را از قبیل لزوم کسب مجوز از شهر با نی جمهوری اسلامی برای گشایش شعب و دفاتر در شهرستانها و یا لزوم کسب مجوز از وزارت ارشاد اسلامی برای چاپ نشریات را در آن آئین نامه بگنجانند. و انجمن های علمی موظف باشند در اساسنامه های خود، فقط یک ماده که نشانگر تبعیت آنها از آئین نامه مزبور باشد را اضافه نمایند. بدین ترتیب، هم تغییر در مواد چندین آئین نامه ای برای مسئولین مربوطه را احتراست و هم انجمن های علمی نیازی به تغییر در اساسنامه خود نخواهند داشت.

سردبیر

گزارش کامپیوتر

ماهنامه انجمن انفورماتیک ایران



زیر نظر کمیته انتشارات انجمن

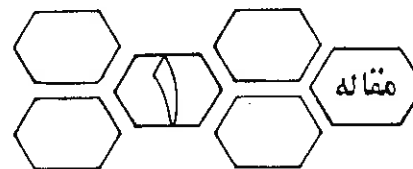
نشانی: صندوق پستی ۱۱۹۶ - ۱۴۱۵۵، تهران، ایران

سرپرست کمیته انتشارات و سردبیر: ابراهیم نقیب زاده، مشایخ

آخرین مهلت قبول مطلب و آگهی: سومین چهارشنبه، ماه قبل از انتشار
نقل مطالب مانده به گزارش کامپیوتر با ذکر مأخذ آزاد است

COMPUTER REPORT, Monthly Publication of the Informatics Society of Iran (ISI)

Address: P.O.Box 14155-1196, Tehran, IRAN--Editor: Ebrahim N. Mashayekh



مروری بر جنبه‌های کامپیوتری طرح دفاع فدوموشکی ایالات متحده آمریکا (موسوم به جنگ ستارگان)

نوشته: دکتر بهروز پرهامی
استاد دانشگاه صنعتی شریف

۱- مقدمه

اینکه بسیاری از پیشرفتهای فنی، بخصوص در زمینه‌های الکترونیک و کامپیوتر، بر اثر طرحهای نظامی و مسابقه تسلیحاتی بین قدرتهای بزرگ جهان ناشی می شوند، واقعیت تأیید باری است که با توجه به نظام سیاسی و اقتصادی حاکم بر جهان امروز چاره‌ای جز پذیرش آن نداریم.

اهمیت روزافزون کامپیوتر در تسلیحات و روشهای نظامی سبب خواهد شد که طی سالهای آینده، سرمایه‌گذاریهای سنگینی در جهت پیشبرد علوم و تکنولوژی کامپیوتر از سوی دستگاههای نظامی ابرقدرتها به عمل آید. بنابر یک نظر:

"کامپیوتر، جنگ امروزی را به یک جنگ اطلاعاتی مبدل ساخته است. دستگاه کوچک و ارزان قیمتی که دارای هوشمندی لازم باشد، می تواند سلاحهای غول پیکر را بی اثر کند. کامپیوتر، داوود را بر جالوت فاتح می سازد."

متخصصان کامپیوتر، مستقل از مواضعی که نسبت به مسابقه تسلیحاتی، ویرانگری وسیع جنگ هسته‌ای، و کاربرد تکنولوژی پیشرفته برای تسهیل قتل عام مردم بیگناه دارند، موظفند که تحولات فنی رشته خود را از هر منبعی که ناشی شوند، دنبال کنند. این مقاله خوانندگان را بابرخی از جنبه‌های کامپیوتری سیستمهای نظامی پیشرفته که طی سالهای باقی مانده از قرن بیستم میلادی و اوایل قرن بیست و یکم در ایالات متحده آمریکا مورد مطالعه قرار خواهند گرفت آشنا می سازد.

بدیهی است که مقالاتی از این دست باید در فواصل زمانی مناسب طی سالهای آینده تهیه شوند تا متخصصان کامپیوتر کشورمان را در جریان آخرین تحولات قرار دهند.

۲- مختصری درباره طرح دفاع فدوموشکی

رقابت تسلیحاتی ابرقدرتهای شرق و غرب تا سال ۱۹۸۳ میلادی بر فلسفه انهدام متقابل مبنی بود. بدین معنی که هریک از دو طرف،

تسلیحات اتمی لازم را برای انهدام کامل طرف مقابل در اختیار داشت و چون پیشقدم شدن در یک جنگ اتمی تمام عیار به معنای انهدام کامل هر دو طرف بود، انگیزه‌ای برای آغاز حمله وجود نداشت.

در سال ۱۹۸۳، طرحی جنجالی از سوی رئیس‌جمهور آمریکا ارائه گردید. برطبق این طرح، باید یک سیستم دفاعی ایجاد شود که بتواند خسارت ناشی از یک حمله تمام عیار اتمی از سوی شوروی را (به شرحی که خواهد آمد) فوق العاده محدود سازد. فلسفه طرح جدید این است که اگر هر دو ابرقدرت دارای این نوع سیستم دفاعی باشند، سلاحهای اتمی به‌طور کلی بی اثر خواهند شد و خطر جنگ اتمی از بین خواهد رفت.

این طرح، که به‌طور رسمی "پیشنهاد دفاع استراتژیک ۵" نامیده می شود، از تکنولوژی پیشرفته فضای سود خواهد جست و به همین دلیل از سوی رسانه‌های هگانی به صورت "جنگ ستارگان" (در تشبیه به نبردهای فضایی تخیلی که در فیلمی به همین نام تصویر شده‌اند) نامگذاری گردید.

پیشنهاد دفاع فدوموشکی به‌طور خلاصه چنین است. در پرتاب کلاهکهای اتمی (اصطلاحی که برای بمبهای اتمی پرتاب شده به کمک موشکهای قاره‌پیما به کار می رود)، چهار مرحله قابل تشخیص هستند:

۱- مرحله برخاست: از هنگام پرتاب موشک تا خروج آن از جو زمین.

۲- مرحله نشانه‌روی: آزاد شدن تدریجی کلاهکهای اتمی و قطعات گلول زن.

۳- مرحله حرکت: حرکت آزاد کلاهکها در فضا به سوی هدفهای تعیین شده.

۴- مرحله فرود: از هنگام ورود کلاهکها به جو زمین تا اصابت به هدف.

بسته به مقرر پرتاب موشک و هدفهای تعیین شده برای کلاهکها، مرحله اول تا ۳ دقیقه، مرحله دوم تا ۵ دقیقه، مرحله سوم تا ۲۲ دقیقه، و مرحله چهارم تا ۱ دقیقه طول می کشد (نمودار ۱).

مرحله برخاست، سفینه حاوی کلاهکها را به فضا می رساند. در مرحله نشانه‌روی، سفینه به کمک موتورهای کوچکی که

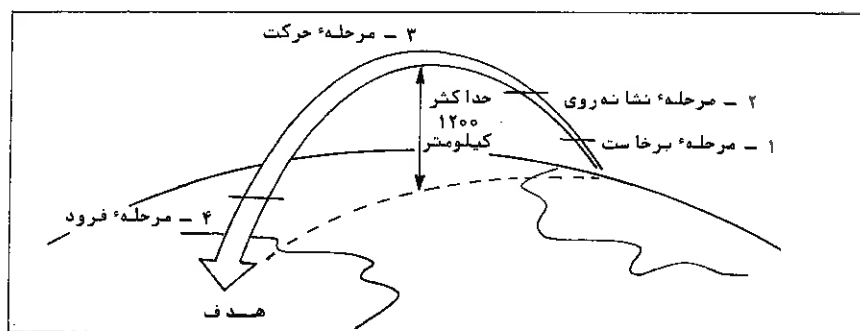
دارد، در فضا تغییر جهت می دهد و پس از هر تغییر جهت، یکی از حداکثر ۱۰ کلاهک اتمی خود (حداکثر تعداد مجاز کلاهکهای اتمی برای هر موشک، برطبق پیمان سال ۱۹۷۲) را همراه با دهها قطعه گلول زن برای گمراه کردن رادارها و دیگر سنجنده‌های دشمن، به سوی هدف مشخصی رها می کند. در مرحله سوم، کلاهکهای سنگین و قطعات سبک گلول زن در فضا با سرعت واحدی به سوی هدف حرکت می کنند. بالاخره، در مرحله چهارم، کلاهکها بر روی هدفها می افتند و قطعات سبکتر از بین می روند.

طرح دفاع فدوموشکی آمریکا که طی ۲۰ الی ۳۰ سال آینده پیاده خواهد شد، سه لایه دفاعی را پیش‌بینی کرده است. لایه دفاعی اول مسئول حمله به موشک در مرحله برخاست است. تأکید سیستم دفاع فدوموشکی بر این است که موشکها را حتی الامکان در همین مرحله از بین ببرد، چون هم تعداد موشکها محدود است (در حال حاضر، حدود ۱۴۰۰ دستگاه) و هم آنکه گرما و نور شدید ساطع شده از موشک، تشخیص و ردیابی آن را توسط سنجنده‌های فضایی به سادگی ممکن می سازند. اما، فرصت دفاع در این مرحله (به دلیل کوتاهی مدت آن) محدود است.

برآورد شده است که اگر شوروی تمام موشکهای خود را طی یک حمله تمام عیار اتمی در فاصله زمانی کوتاهی پرتاب کند، سیستم دفاعی باید بتواند انهدام ۱۰ موشک را در هر ثانیه داشته باشد. سلاحهای پیش‌بینی شده برای این منظور عمدتاً از نوع لیزری هستند که در ایستگاههای فضایی نصب خواهند شد. برای کسب اطلاعات بیشتر درباره این سلاحها می توانید به دومین مقاله از ماخذ شماره ۱ مراجعه کنید.

در لایه دفاعی دوم، کلاهکهای اتمی پس از آزاد شدن و حین حرکت به سوی هدفها مورد شناسایی و حمله قرار می گیرند. کار در این مرحله به دلیل تعدد هدفها (کلاهکهای اتمی و گلول زنها) و نیز لزوم دقت بیشتر در ردیابی آنها، به مراتب مشکلتر است.

ورق بزنید



نمودار ۱ - مراحل چهارگانه پرتاب سلاحهای اتمی بین قاره‌ای.

لایه دفاعی سوم بر روی مرحله فرود کار می کند و کلاهک اتمی را پس از وارد شدن به جو زمین مورد حمله قرار می دهد.

اگر هریک از سه لایه ۹۰ درصد موثر باشد، به فرض پرتاب تمام ۱۴۰۰ موشک کنونی شوروی به طور همزمان، تنها ۱۴۰ تای آنها از لایه دفاعی اول خواهند گذشت. این موشکها، حداکثر ۱۴۰۰ کلاهک اتمی را همراه با تعداد کثیری قطعات گول زن به سه هدفهای مختلف را خواهند ساخت. حدود ۱۴۰ کلاهک از لایه دفاعی دوم می گذرند و بالاخره، ده درصد این تعداد، یعنی ۱۴ کلاهک، به هدفهای خود خواهند خورد.

به فرض افزایش تعداد موشکهای شوروی به ۴۰۰۰، حدود ۳۰ کلاهک از این سیستم دفاعی عبور خواهند کرد. البته همین ۱۴ یا ۳۰ کلاهک نیز قدرت تخریبی فوق العاده ای دارند، اما با توجه به آنکه برای شوروی قابل پیش بینی نیست که کدام کلاهکها به هدف خواهند خورد و قدرت نظامی آمریکا برای حمله، متقابل تا چه اندازه کاهش خواهد یافت، بعید می نماید که در حمله اتمی پیشقدم شود (حداقل مدافعان این سیستم چنین فکر می کنند!).

البته توانائی انهدام ۹۰٪ از هدفها در هر لایه به سادگی قابل حصول نیست و این امر یکی از استدلالهای اساسی مخالفان این طرح را تشکیل می دهد. مثلاً اگر توانائی انهدام هر لایه دفاعی فقط ۷۰٪ باشد، با توجه به ارقام بالا، لایه دفاعی دوم با ۴۲۰۰ کلاهک (۹۰۰۰ کلاهک، اگر تعداد موشکها به ۳۰۰۰ برسد) مواجه خواهد بود که رادیابی و تشخیص آنها از دهها هزار قطعه گول زن کار بسیار مشکلی است. ضمناً، در نهایت حدود ۳۸۰ کلاهک اتمی (۸۱۰ کلاهک) به هدف خواهند خورد.

برای اجتناب از طولانی شدن بحث، در این مقاله چیزی درباره جنبه های سیاسی طرح دفاع ضد موشکی و استدلالهای موافقان و مخالفان آن (مگر در مورد مسائل کامپیوتری) نخواهیم گفت. ماخذ حاوی جدول مقایسه ای جالبی از نظرات موافقان و مخالفان این طرح و موضع گیریهای آنها در مقابل استدلالهای یکدیگر است.

۳ - مشکلات طرح دفاع ضد موشکی

آیسما یک سیستم دفاعی هوشمند با سنجنده های پیچیده و سلاحهای غیر هسته ای می تواند جلوی تخریب وسیع موشکهای اتمی بین قاره ای را بگیرد؟ این سوالی است که طرح دفاع ضد موشکی آمریکا در اذهان مطرح ساخته است. پژوهشهای مقدماتی که ۳۰ میلیارد دلار هزینه خواهند داشت در اوایل دهه ۱۹۹۰ میلادی پاسخ این سوال را مشخص خواهند کرد. هرگاه پاسخ سوال بالا مثبت

باشد، حدود یک هزار میلیارد دلار دیگر صرف بررسی، طراحی، ساخت، و نصب این سیستم دفاعی خواهد شد که هزینه نگاهداری سالانه آن پس از طراحی و نصب بین ۵۰ تا ۲۰۰ میلیارد دلار برآورد شده است.

برای عملی شدن این طرح، کامپیوترهای پُر قدرت بلا درنگ با قابلیت تحمل عیوب و خرابیها مورد نیازند تا عملیات تجسس، جمع آوری داده ها، رادیابی، هدف گیری، و ارزیابی خسارت وارده به کلاهکها را تحت شرایط مشکل و غیر قابل پیش بینی یک جنگ تمام عیار اتمی به دقت انجام دهند.

برخی بر این عقیده اند که طراحی و ساخت سخت افزار و نرم افزار کامپیوتر در واقع مشکلترین بخشهای این طرح را تشکیل می دهند، چون به نظر می رسد که سلاحها و ایستگاههای فضائی مورد نیاز با مطالعه کافی و صرف بودجه ای هنگفت قابل ساخت هستند. البته این بدان معنی نیست که مسائل فضائی و تسلیحاتی طرح به طور کامل حل شده اند، بلکه تنها بر این موضوع دلالت دارد که تا میسر هوشمندی و قابلیت اطمینان مورد نیاز برای سیستمهای کامپیوتر این طرح، فوق العاده مشکل خواهد بود.

اهم مشکلات فنی قابل پیش بینی برای این طرح عبارتند از:

- ۱ - یافتن روشهای رادیابی و تشخیص کلاهکهای اتمی از قطعات گول زن، با توجه به تعداد قطعات شناور در فضا (جمعاً) تا حدود ربع میلیون هدف مختلف).
- ۲ - لزوم از بین بردن اکثر موشکهای دشمن در مرحله برخاست، با توجه به کوتاه بودن زمان موجود و عدم امکان دخالت دادن عوامل انسانی در تصمیم گیری.
- ۳ - طراحی نرم افزار قابل اطمینان برای هدایت عملیات، با توجه به آنکه این نرم افزار باید بتواند مدها هزار تصمیم گیری مستقل را ظرف چند میلی ثانیه هماهنگ کند.
- ۴ - حفاظت خود سیستم دفاعی در مقابل حملات دشمن.

۵ - ساخت و نصب سیستم با هزینه های نه چندان زیاد، به طوری که دشمن انگیزه اقتصادی برای بی اثر ساختن آن از راه افزایش سلاحهای تهاجمی خود نداشته باشد.

در زمینه تعداد ماهواره ها یا ایستگاههای فضائی رادیابی و انهدام موشکها و کلاهکها، برآوردهای مختلفی موجودند که با فرضهای گوناگون در مورد توانائیهای تسلیحات فضائی، به دست آمده اند. پائین ترین برآورد حدود ۳۰ ایستگاه فضائی را پیش بینی می کند و بر دوفرض اساسی مبتنی است:

- ۱ - امکان ساخت سلاحهای فضائی بسیار پیشرفته و بخصوص لیزرهای با قدرت فوق العاده زیاد.

۲ - آسیب نا پذیر بودن سیستم دفاعی در مقابل حملات دشمن.

بالاترین برآوردها وجود چند هزار ایستگاه فضائی را برای تأمین توانائیهای مورد نظر لازم می دانند.

واضح است که این برآوردها، و نیز ارقام دیگری که درباره این طرح انتشار می یابند، باید با احتیاط کامل تلقی شوند، زیرا ارقام و برآوردهای واقعی بخشی از اسرار نظامی را تشکیل می دهند و مسلماً "به سادگی فاش نخواهند شد."

۴ - نیازهای سخت افزاری و نرم افزاری

تشخیص و رادیابی موشکها و کلاهکها بر عهده ماهواره های تجسسی است. اگرچه رادیابی یک موشک بین قاره ای و انهدام آن در فضا توسط یک موشک دفاعی از سال ۱۹۶۲ میلادی عملی بوده است، اما رادیابی هزاران قطعه شناور در فضا (کلاهکهای واقعی و گول زنها) و تخصیص امکانات دفاعی به آنها به صورت بهینه (یعنی تصمیم گیری در مورد اینکه هر قطعه "خطرناک" باید توسط کدام ایستگاه فضائی یا چگونگی سلاخی منهدم شود) باروشهای امروزی امکان پذیر نیست.

یک روش پیشنهادی برای این منظور چنین است. هر ماهواره تجسسی مجهز به یک صفحه حساس ۵۰ میلیون عنبری است که هر عنصرش اطلاعات دریافتی از فضائی متناظر با یک کیلومتر مربع از سطح کره زمین را در یک کلمه ۱۰ بیتی خلاصه می کند. اگر برای حصول دقت لازم، دو نمونه برداری در ثانیه لازم باشد، حجم داده های مورد پردازش به یک گیگا بیت (هزارمگا بیت) در ثانیه بالغ می شود. با توجه به نوع پردازشهای اعمال شده بر روی این داده ها، قدرت محاسباتی ۱۰ گیگا عمل ممیز شناور در ثانیه^۷ (یعنی حدود ۱۰۰۰ بار بیشتر از کامپیوترهای موجود در سفینه های فضائی امروزی) مورد نیاز خواهد بود.

برای تأمین این قدرت محاسباتی، دوروش به نظر می رسند. در روش اول، که نیازمند پژوهشهای کاملاً جدید طی سالهای آینده است، سرعت سخت افزار کامپیوتر با استفاده از تکنولوژیهای نوین (مثلاً روشهای نسوری) افزایش می یابد. در روش دوم، که از تکنولوژی متعارف استفاده می کند، محاسبات مورد نظر بین پردازشگرهای متعدد توزیع می شوند. برای استفاده از روش اخیر، مسائل متعددی باید از راه پژوهشهای دراز مدت حل شوند. مثلاً شبکه های مرتبط سازنده پردازشگرها باید طوری ساخته شوند که حتی با ازکارافتادن برخی اتصالات، بتوانند وظایف محوله را به درستی انجام دهند.

البته برخی از شبکه های کامپیوتری امروزی می توانند در صورت خرابی گره ها یا



اتصالات، پیگیربندی خود را تغییر دهند و مسیرهای جدید مناسب برای ارسال پیامها را بیابند. اما زمان لازم برای تغییر پیکر بندی در این شبکهها به چند ثانیه می رسد و بنابراین، روشهای موجود برای یک جنگ تمام عیار فضا ئی، که ممکن است طی آن مدها تغییر پیگیربندی در هردقیقه لازم باشند، قابل استفاده نیستند. به علاوه، امکان بروز عیبها یا خرابیهای جدید در حین تغییر پیگیربندی، مسأله را به مراتب مشکل تر می کند.

کامپیوترهای مورد نیاز برای این طرح باید به مدت ۱۰ سال بدون نیاز به تعمیر و نگهداری به درستی کار کنند. برای مقایسه، بدنست بدانید که کامپیوترهای متعارف امروزی، باید هر چند روز یک بار به حالت اولیه بازگردانده شوند و کار خود را از سر بگیرند و هر چند ماه یک بار نیز نیاز به تعمیر و تعویض قطعات سخت افزاری دارند.

به فرض تأمین قدرت سخت افزاری مورد نیاز با قابلیت اطمینان کافی، مشکل طراحی نرم افزار هنوز باقی است. برآورد اولیه اندازه نرم افزارهای مورد نیاز برای این سیستم دفاعی، حدود ۱۰ میلیون سطر (حکم) بوده است، اما احتمال دارد که اندازه واقعی تا ۱۰ برابر بیشتر شود. برای هر کسی که در طراحی یک سیستم واقعی نرم افزاری مشارکت کرده باشد، اندازه این نرم افزارها واقعا " وحشتناک است. شاید استغفای برخی از متخصصان طراز اول نرم افزار، که در رابطه با این طرح به تحقیق مشغول بودند، از همین امر ناشی شده باشد.

برخی از متخصصان نرم افزار برای این عقیده اند که بدلیل نامشخص بودن پارامترهای سیستم مورد نیاز، طراحی نرم افزار برای آن ناممکن است. این عده چنین استدلال می کنند که شوری در مقابل طراحی ایسی سیستم ساکت نخواهد نشست و سعی خواهد کرد تا مشخصات سلاحهای تهیجی خود را مرتباً تغییر دهد. هر چه کار طراحی سیستم دفاعی جلوتر برود، سرعت تغییرات بیشتر خواهد شد و در نهایت، کا بوسی وحشتناک را برای دوطرف درگیر و بقیه دنیا به وجود خواهد آورد.

از طرف دیگر، نرم افزار این سیستم هرگز پیش از حمله واقعی مورد آزمایش کامل قرار نخواهد گرفت و وجود اشتباههای متعدد در آن (ناید تا دهها هزار اشتباه) اجتناب ناپذیر است. بنابراین، نرم افزار سیستم باید طوری طراحی شود تا عیبها و اشتباهات موجود را (چه در خود نرم افزار و چه در سخت افزار پشتیبان آن) تحمل نماید و این امر بر پیچیدگی کار می افزاید.

با توجه به مشخصات سخت افزار و نرم افزار این سیستم دفاعی، طرحهای پژوهشی

متعددی در زمینه های زیر در ایالات متحده آمریکا به جریان افتاده اند:

- ۱ - پردازش موازی.
 - ۲ - کامپیوترهای نوری.
 - ۳ - شبکه های کامپیوتری با اتصال کامل بین هر دو نقطه دلخواه.
 - ۴ - هوش مصنوعی و تصمیم گیری خودکار.
- سه زمینه اول مستقیماً از مسائل کلیه برشمرديم ناشی می شوند، اما زمینه چهارم نیازمند توضیح بیشتری است.
- تأکید این سیستم دفاعی بر رشته هوش مصنوعی به دو دلیل عمده قابل توجیه است. اول آنکه تشخیص کلاهکهای اتمی واقعی از قطعات گول زن، آن هم از فاصله بسیار دور، نیازمند روشهای دقیق تصویربرداری، الگوسازی^۸، و تصمیم گیری بر مبنای معلومات کسب شده است. دوم آنکه از بین بردن موشکها در مرحله برخاست (فرصتی حداکثر تا ۳ دقیقه) نیازمند تصمیم گیری خودکار و بدون دخالت عوامل انسانی است. اگر سیستم دفاعی به اشتباه فعال شود، ایمنی احتمال وجود دارد که تصور حمله در دشمن ایجاد گردد و یک جنگ اتمی ناخواسته به راه بیافتد.

سیستمهای خبره^۹، که در بحثهای امروزی رشته هوش مصنوعی جای ویژه ای دارند، در این سیستم دفاعی نیز نقش حساسی را ایفا می کنند. اما یک مشکل اساسی در زمینه طراحی این سیستمهای خبره وجود دارد. سیستم های خبره ای که تاکنون ساخته شده اند همگی تقلیدی از انسانهای خبره بوده اند. به عبارت دیگر، معلومات و تجربیات یک یا چند انسان خبره به صورت داده ها و قوانینی در سیستمهای خبره کامپیوتری تعبیه می شوند. اما در مورد دفاع ضد موشکی، انسانهای خبره ای وجود ندارند که با تمام زوایای مسأله آشنا باشند (چون چنین جنگی هرگز اتفاق نیافتاده است). یافته های سالیهای اخیر نشان می دهند که سیستمهای خبره کنونی در مواجهه با وضعیتهای غیرمنتظره یا خارج از دامنه معلومات و تجربیاتشان، به نحوی مضحک رفتار می کنند و این امر در یک سیستم دفاع ضد موشکی قابل تحمل نیست. در حال حاضر، ساخت یک مرکز آزمایش سیستمهای داده پردازی و مخابراتی، که توانائی شبه سازی یک جنگ اتمی را نیز داشته باشد، در دست بررسی است. انجام این گونه شبه سازیها هزینه ای سنگین (شاید بیش از هزینه اصل سیستم) در بر خواهد داشت و حصول اطمینان از درستی نرم افزارهای شبه سازی، با توجه به نادقیق یا تخمینی بودن برخی از داده های مورد نیاز، خود مسأله بسیار پیچیده ای است.

۵ - پایگاه داده های توزیع شده^{۱۰}

برای تعبیر و تفسیر داده ها توسط

کامپیوتر ماهواره، باید یک پایگاه داده های کامل از مشخصات سلاحهای مختلف موشکی (به صورتی که توسط سنجنده ها " دیده " می شوند) ایجاد گردد. این پایگاه داده ها همچنین برای تشخیص دادن کلاهکهای واقعی از گول زنها و قطعات ازمه پاشیده موشک و مخازن سوخت آن مفید است. چون همراه با هر کلاهک واقعی، دهها قطعه گول زن در فضا رها می شوند، عدم تشخیص درست کلاهکهای واقعی از گول زنها ممکن است باعث به هدف رسیدن کلاهکها یا اتلاف امکانات (وقت، انرژی، و تسلیحات) برای از بین ببردن قطعات بی زبان گردد.

چون احتمال می رود که برخی از پایگاه های فضا ئی در حین جنگ منهدم شوند، داده های مورد نیاز برای ارزیابی وضعیت و تصمیم گیریها باید به صورت توزیع شده و افزونه^{۱۱} نگاه داری شوند. مشکل اساسی در مورد وجود اطلاعات افزونه، سازگار نگاه داشتن نسخه های مختلف داده ها علیرغم بروز نواقص و خرابیها است. در حال حاضر، معلوم نیست که بتوان چنین پایگاه داده های توزیع شده ای را که نواقص و خرابیها را تحمل کند و از سوی منابع متعدد نیز قابل دسترسی و به هنگام سازی باشد طراحی نمود.

مشکل دیگر، حجم عظیم داده ها و تراکتهای^{۱۲} وارد شده به سیستم است. بزرگترین پایگاههای داده امروزی، که عمدتاً برای بانکهای بزرگ طراحی شده اند، می توانند تا ۱۰۰۰ تراکنش را در ثانیه پاسخگو باشند. برای سیستم دفاعی پیشنهادی، تعداد تراکنشها ۱۰ برابر یا بیشتر خواهد بود.

در واقع، سیستم دفاع ضد موشکی دارای دو پایگاه داده متمایز به شرح زیر است:

- ۱ - پایگاه داده های مربوط به تهیج: مختصات فضا ئی، مسیر حرکت، و هویت تمام موشکها، سفینه های رها شده، کلاهکهای اتمی، گول زنها، و قطعات متلاشی شده شناور در فضا. یک مسأله مهم در مورد این پایگاه داده ها، در نظر گرفتن روشهای ترمیم برای مواردی است که سنجنده های مختلف، اطلاعات ناسازگاری را ارائه می دهند.
- ۲ - پایگاه داده های مربوط به دفاع: سالم بودن یا نبودن پایگاههای فضا ئی، کامپیوترها، و سنجنده ها، مختصات پایگاههای فضا ئی، و میزان مهتات باقی مانده هر پایگاه. این پایگاه داده ها باید طوری طراحی شود که به سادگی بتوان در صورت انهدام یک یا چند پایگاه فضا ئی، هدفهای را که قبلاً نابودیشان به این پایگاهها محول شده بود بین پایگاههای فضا ئی دیگر توزیع نمود (تجدید اختصاص منابع).

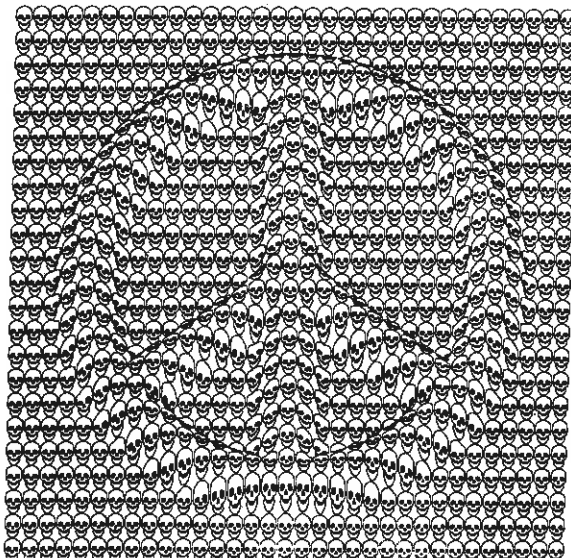
هر دو پایگاه داده ها، به دلیل افزونه بودن اطلاعات موجود در آنها، ممکن است حاوی ورق بزنند

۳ - شناسایی زبان طبیعی.

مورد اول در هر سه طرح کاربرد دارد و موارد دوم و سوم برای برقراری ارتباط طبیعی و سریع بین انسان و ماشین در شرایط خاص میدان نبرد ضروری هستند.

مآخذ و یادداشتها

1. Adams, J.A., M.A. Fischetti, J. Horgan, and P. Wallich, "Star Wars --SDI: The Grand Experiment," شماره مخصوص مجله IEEE Spectrum یک مقدمه و ۲ مقاله مستقل با عناوین "Mind-Boggling Complexity," "Exotic Weaponry," and "Debating the Issues" Vol. 22, No. 9, pp. 34-64, Sep. 1985.
2. Stefik, M., "Strategic Computing at DARPA: Overview and Assessment," Communications of the ACM, Vol. 28, No. 7, pp. 690-704, July 1985.
3. Defense Advanced Projects Research Agency, "Strategic Computing -- New Generation Computing Technology: A Strategic Plan for Its Development and Application to Critical Problems in Defense," DARPA Report, Reprinted in Computer Architecture Technical Committee Newsletter, IEEE Computer Society, pp. 88-176, Mar. 1985.
4. "Software for Star Wars 'Impossible Task', Say Experts," Computing, 5 Sep. 1985.
5. SDI = Strategic Defense Initiative
6. Real-time
7. Giga FLOPS = 10^9 Floating-point Operations Per Second
8. Pattern recognition
9. Expert systems
10. Distributed database
11. Redundant
12. Transactions
13. Asynchronous
14. DARPA = Defense Advanced Research Projects Agency
15. Computer Bulletin, Sep. 1985.



کامپیوترها، با تواناییها و هوشمندی بسیار زیاد، کاربردهای نظامی نوینی را ممکن می سازند. برای فراهم ساختن قالبی که پژوهشها را منسجم و جهت دار کنند، دارپا فعالیت در زمینه طراحی سیستم خاص را توصیه کرده است:

۱ - خودروی زمینی هوشمند، با توانایی تشخیص موانع و حرکت خودکار در مسیرهای نامواری برای انجام عملیات تجسس، پشتیبانی، و مهمات رسانی در نواحی خطرناک.

۲ - کمک خلبان کامپیوتری که به عنوان دستیار توسط خلبان تعلیم داده می شود و در موارد اضطراری (مثلاً) در هنگام بیهوش شدن خلبان بر اثر شتاب زیاد ناشی از مانورهای سریع برای فرار از اصابت موشک) هدایت هواپیما را به دست می گیرد.

۳ - سیستم مدیریت رزمی برای ناوهای هواپیما بر که بتواند تصویر دقیق منطقه نبرد را همراه با وضعیت نیروهای دو طرف و شرایط جوی به نمایش بگذارد، فرضیه‌های در زمینه نتایج دشمن اراکه دهد، و نتایج حاصل از تصمیمات نظامی گوناگون را از راه شبیه سازی با یکدیگر مقایسه کند.

وجه اشتراک این سیستم، نیاز آنها به استفاده از یک سیستم خبره بسیار پیچیده برای امکان پذیر ساختن استنتاجها و تصمیم گیریها است. این سیستمهای خبره، به دلیل عملکرد در محیط بلادرنگ، نیاز به قدرت پردازش بسیار زیاد دارند و بنابراین، پژوهش در زمینه تکنولوژی سخت افزار و معماری کامپیوتر را به عنوان فعالیتی زیربنایی مطرح می سازند.

زمینه های دیگری که باید برای تحقق یافتن این سیستمها بررسی شوند عبارتند از:

- ۱ - پردازش و شناسایی تما و بیضه (بینائی مصنوعی).
- ۲ - تولید و شناسایی کلام (گویائی و شنوای مصنوعی).

اطلاعات ناسازگار باشند. به عنوان مثال، ممکن است که یک نسخه از اطلاعات به هنگام درآمده باشد، اما هنوز نسخه های دیگر به صورت قبلی باقی مانده باشند. اگر این ناسازگاری به دلیل بروز خرابی در بخشهای از سیستم، دائمی شود، مسلماً "خطرناک" خواهد بود. اما حتی ناسازگاری لحظه ای ناشی از به هنگام درآوردن نسخه های مختلف اطلاعات به صورت نا همگام ۱۳ نیز می تواند، به دلیل سرعت زیاد تراکنشها و تصمیم گیریها، مشکلاتی را به وجود آورد. این نوع عملکرد نا همگام در پایگاههای داده های توزیع شده هنوز چندان برای پژوهشگران این رشته آشنا نیست.

۶ - پژوهشهای بنیادی در رشته کامپیوتر

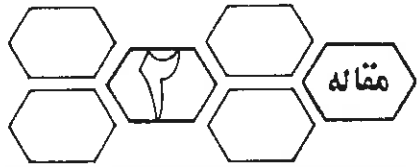
وزارت دفاع ایالات متحده آمریکا دارای بخشی به نام "دارپا" ۱۴ است که پژوهشهای بنیادی را در رشته کامپیوتر پشتیبانی می کند. اگرچه اغلب طرحهای پژوهشی پشتیبانی شده توسط دارپا جنبه گسترش مرزهای دانش را دارند، اما انتخاب طرحها مسلماً جهت دار و در خدمت هدفهای نظامی است. مثلاً در طرح ۱۰ ساله ای که اخیراً دارپا با بودجه ای متجاوز از یک میلیارد دلار پیشنهاد کرده است، تأکید بسیار زیادی بر کامپیوترهای هوشمند و سیستمهای خبره وجود دارد؛ یعنی درست همان چیزهایی که در طرح دفاع ضد موشکی مورد نیازند. بودجه تخصص یافته از سوی دارپا به این طرح در سال ۱۹۸۴ برابر ۵۰ میلیون دلار بود. این بودجه به تدریج افزایش می یابد و در سال ۱۹۸۸ به ۱۵۰ میلیون دلار می رسد (جمع کل بودجه برای ۵ سال اول = ۵۲۰ میلیون دلار). برای ۵ سال بعد، بودجه دقیقتری مشخص نشده است و مبالغ تخصیص یافته به میزان پیشرفت کارها در ۵ سال اول بستگی خواهند داشت. اگرچه این بودجه چیزی در حدود ۲٪ مبلغی است که برای طرح دفاع ضد موشکی طی پنج سال آینده صرف خواهد شد، اما با توجه به آنکه پژوهشهای بنیادی عمده در دانشگاهها با استفاده از نیروی کار ارزان قیمت دانشجویان به اجرا درمی آید، آثار حاصل از آن مسلماً چشمگیر خواهند بود.

بسیاری از ناظران، این طرح را پاسخی به پروژه کامپیوترهای نسل پنجم زاپن تلقی کرده اند. اما علیرغم برخاستن از شباهتها (مثلاً میزان بودجه، پیشنهادها، تأکید بر هوش مصنوعی، تلاش در جهت افزایش سرعت کامپیوترها)، تفاوتها زیادی نیز از نظر هدفها بین دو طرح موجودند. مثلاً دارپا ابتدا "هدفهای نظامی خود را از اجرای این طرح پنهان نمی کند و در مقدمه طرح به این نکته اشاره دارد که نسل جدید

نمودار ۲ - این تصویر از

علامت "صلح"، که به کمک کامپیوتر رسم شده است، ۱۵، نظر یک تصویر ساز (گرافیکست) کامپیوتری را درباره چشم انداز صلح جهانی در عصر تسلیحات اتمی و جنگ افزار الکترونیکی نشان می دهد.

مقاله



آی بی ام
به آزمایشگاههای تکنولوژی پیشرفته، ژاپن
راه می‌یابد

ترجمه: محمد میرزا عبداللہی

در حدود ۲۵ سال پیش، ژاپن به شرکت آی بی ام فشار آورد تا اختراعات ثبت شده خود را در ازای بدست آوردن حق تولید آنها در داخل ژاپن، در اختیار کمپانیهای ژاپنی قرار دهد. بدین ترتیب آی بی ام کمک کرد تا پایه‌های تشکیل صنعت کامپیوتر در ژاپن ریخته شود و این صنعت به نوبه خود، تا شورشگفت انگیزی در بازار کامپیوتر به وجود آورد. اکنون، در بخش تاریخ، ژاپنیها با سرعت بسیار، تکنولوژی پیشرفته و مستقل خود را به وجود می‌آورند و حال این آی بی ام است که در خانه ژاپن را می‌زند. بر اساس یک موافقت نامه ۵ ساله که در تابستان سال به امضاء می‌رسد، آی بی ام حق خواهد داشت از نتایج تحقیقاتی که توسط محققین ژاپنی در زمینه کامپیوتر انجام می‌پذیرد، بهره‌مند گردد. هم اکنون در مرکز ثبت اختراعات ژاپن، بیش از ۴۳۰ اختراع، شامل مدارهای مجتمع سه بعدی که مقدار بسیار زیادی اطلاعات را در خود ذخیره می‌نمایند، مدارهای گالیم آرسناید^۱ که بسیار سریعتر از مدارهای سیلیسیمی فعلی هستند، و سیستمهای نوری^۲ که به عنوان چشم در ماشینهای صنعتی کارخانجات بکار گرفته می‌شوند، ثبت گردیده است. به علاوه، آی بی ام قادر خواهد بود به اختراعاتی که در پروژه ۱۰ ساله و ۱۰۰ میلیون دلاری "برکا میپوتر"^۳ (که هم اکنون در ژاپن در جریان است) حاصل خواهد شد، دست یابد. همچنین، در صورتی که پروژه ۲۵۰ میلیونی دلاری ساخت کامپیوترهای نسل پنجم با استفاده از هوش مصنوعی نتایج داشته باشد، آی بی ام از نتایج آن بی‌بهره نخواهد ماند.

عواقب قرارداد بین آی بی ام و ژاپن که اطلاعات آن در ماه جولای سال جاری مسیحی در توکیو در زبیدا کرد، می‌تواند بسیار مهم باشد. اول این که معلوم نیست که رقابتی آی بی ام نیز از این نتایج بهره‌ای می‌گیرند یا خیر. با وجودی که آی بی ام ادعا می‌کند که دستیابی به اختراعات انحصاری نیست، اما بسیاری از کمپانیهای آمریکایی توانائی و نفوذ کافی برای انعقاد قرارداد مشابهی را با ژاپن ندارند. و این در حالی است که آی بی ام نیز خود چندین سال برای انعقاد این قرارداد تلاش کرده است.

به عبارت دیگر، چنانچه آی بی ام بزرگ (کنایه

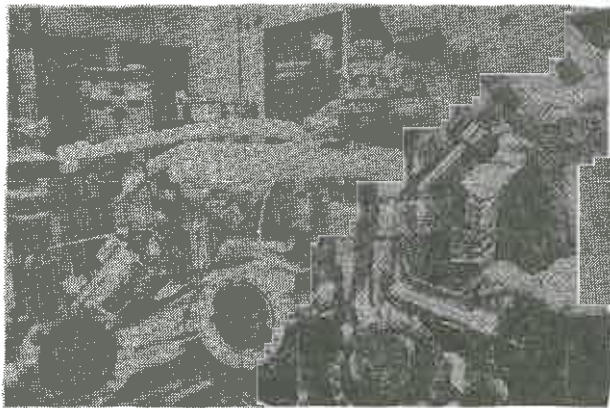
از آرم آی بی ام است) برای یکی از نتایج تکنولوژی ژاپن بازاری بیابد، رقابت قدرت رقابت در آن زمینه را نخواهند داشت. ونیز این که آی بی ام می‌تواند وارد رشته‌هایی از قبیل سیستمهای نوری و برکا میپوترها شود و با پیشتر از آن ماندمرکز تحقیقات "گری"^۴ و شرکت تلفن و تلگراف آمریکا^۵ رقابت نماید. دست اندرکاران صنعت کامپیوتر ژاپن نیز تصدیق می‌نمایند که کمپانیهای دیگر به آنچه که آی بی ام بدان دست یافته، به سادگی دست نخواهند یافت. به گفته یکی از مقامات ارشد "بنگاه علم و تکنولوژی صنعتی"^۶ ژاپن، "صنعت پردازش داده‌ها، صنعت خاصی است. در اصل، تقاضای برای دستیابی به تکنولوژی هنوز به صورت تک تک بررسی می‌شود و جهت انتقال اختراعاتی که به وسیله چند مرکز به نتیجه رسیده باشد، موافقت تمام آنها الزامی است."

حتی اگر دیگر تولیدکنندگان کامپیوتر نیز بعد از آی بی ام به تکنولوژی ژاپن دست

به یک موازنه رسیده ایم. آنها به اختراعات ما دسترسی داشتند ولی ما از دستیابی به اختراعات آنها محروم بودیم."

واضح است که هنوز محدودیتهایی در امر انتقال تکنولوژی ژاپن به آی بی ام و بسیاری از شرکت‌های خارجی دیگر وجود خواهد داشت. بعضی از متخصصین اظهار می‌نمایند که کمپانیهای ژاپنی بعضی از اطلاعات را مخفی نگاه خواهند داشت (به ثبت نمی‌رسانند) تا رقابتی خارجی از آنها بهره‌مند نگردد. ولی قرارداد آی بی ام ممکن است باعث شود تا دولت ژاپن مجبور به داشتن سیاستهای بازتری در رابطه با پیوندهای تحقیقاتی با آمریکا گردد. اگر این امر تحقق پذیرد، مسیر مبادله تکنولوژی بین ژاپن و آمریکا، مسیر دوطرفه‌ای خواهد شد.

به نقل از
BUSINESS WEEK
AUGUST 1985



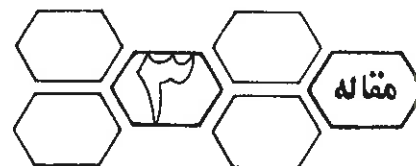
یادداشتها

- 1: Gallium Arsenide Circuits
- 2: Optical Systems
- 3: Supercomputers
- 4: Cray Research Inc.
- 5: American Telephone & Telegraph Co. (AT&T)
- 6: Agency of Industrial Science & Technology
- 7: Ministry of International Trade & Industry (MITI)
- 8: Biotechnology



یابند امکان دستیابی به روشهای ساخت در رشته‌ای دیگر کمتر خواهد بود. مسئولین وزارت صنایع و تجارت بین المللی ژاپن^۷ اعلام داشته اند که بسیاری از کمپانیهای ژاپنی که در امر تحقیقات در زمینه‌های مختلف، از ساجی گرفته تا تکنولوژی زیستی^۸، فعال بوده‌اند، سرخشان به انتقال تکنولوژی به کمپانیهای آمریکا می‌خافند.

آی بی ام می‌گوید این بسیار منصفانه است که به اختراعات ژاپنیها دست یافته است. امر مبادله اختراعات بین آی بی ام و کمپانیهای تولیدکننده کامپیوتر ژاپن به مدت ۱۰ سال در جریان بود. ولی به تازگی وزارت صنایع و تجارت بین المللی ژاپن پس از سرمایه‌گذاری در یک زمینه مهم تحقیقاتی، اعلام نموده بود که نتایج این تحقیقات به آن وزارتخانه تعلق دارد و در دسترس دیگران قرار نخواهد گرفت. با این وجود، دولت آمریکا در این مدت اختراعات غیردفاعی خود را مجاناً در اختیار ژاپن قرار داده است. یکمی از مدیران آی بی ام در نیویورک می‌گوید: "ما



می‌شوند. یکی از انواع متداول آنها رویه‌هایی است که متغیرهای برنامۀ فراخواننده را تغییر می‌دهند. بدین مثال در زبان پاسکال توجه کنید:

```
procedure GETRS(X,Y:real);
  RS*/ در بلوک خارجی اعلان شده است */
begin RS:=X*Y+Y*Y
end;
```

نبودن متغیرهای مشترک یا سراسری و کنترل دقیق حوزه‌های متغیرها، کامپایلرها قادر می‌سازد که مانع این نوع از اثرات جنبی گردد. اما یک کامپایلر متغیرهای برجرایان داده‌ها، ممنوعیت‌های بیشتری در مورد اثرات جنبی اعمال می‌کند: یک رویه، حتی شناسه‌های خود را نیز نمی‌تواند تغییر بدهد. در حقیقت، هیچ چیز، هیچگاه تغییر نمی‌کند.

برای تعیین ممنوعیت‌های لازم در مقابل اثرات جنبی، به منظور حصول همزمانی در محاسبات، برنامۀ‌ها را به شکلی داده‌ای مانند آرایه‌ها و رکوردها را دستکاری می‌کنند، مورد بررسی قرار می‌دهیم.

این رویه را که با استفاده از مکانیزم "فراخوانی با ارجاع"، شناسه‌های خود را تغییر می‌دهد، در نظر بگیرید:

```
procedure SORT2(var A:array[1..10] of
  real;J:integer);
  var T:real;
  begin if a[J] > A[J+1] then begin
    T:=A[J];
    A[J]:=A[J+1];
    A[J+1]:=T;
  end
end;
```

SORT2، رویه‌ای است که دو عنصر J ام و J+1 ام آرایه A را با تعویض آن دو در صورت لزوم به صورت صعودی مرتب می‌کند. اکنون به این احکام توجه کنید:

(۱) SORT2(AA,J);
(۲) SORT2(AA,K);
(۳) P:=AA(L);

احکام (۱) و (۲) و (۳) ممکن است با یکدیگر تداخل داشته باشند. چون مقادیر J و K و L برای کامپایلر مشخص نیست، باید فرض کند که این احکام ممکن است به یکدیگر وابسته باشند و آن‌ها را درست به همان ترتیب مشخص شده، اجرا کند. هر تلاشی برای اجرای همزمان این احکام ممکن است منجر به آلودگی نامدن نتایج نام درست شود (بسته به این که مقدار I و J و K چه باشد). بدین مثال توجه کنید:

```
procedure SORTREAD(var A,B:array[1..10]
  of real;I,J:integer);
  SORT2 قبل تعریف شده است */
begin SORT2(A,I);
  RESULT:=B[J];
end;
```

تکنولوژی ماساچوست و دانشگاه کالیفرنیا در ایروین^۷ ابداع شده‌اند. زبان "لوسید"^۸ در اصل برای بازیابی برنامۀ‌ها - و به منظور برنامۀ‌سازی برای کامپیوترهای مبتنی بر جریان داده‌ها - طراحی شده است و زبان مناسی برای محاسبات مبتنی بر جریان داده‌ها نیست. بطور کلی، هر زبان برنامۀ‌سازی برای یک کامپیوتر مبتنی بر جریان داده‌ها، باید دو ضابطه اساسی زیر را برآورده سازد:

- ۱ - قادر به استنباط وابستگی داده‌ها در عملیات مختلف برنامۀ‌ها باشد.
- ۲ - محدودیت‌های ترتیب‌گذاری آن، دقیقاً مطابق با وابستگی داده‌ها باشد، یا به عبارتی دیگر، اجرای هر دستورالعمل تنها مبتنی بر دسترس پذیری داده‌ها باشد.

تأثیر موضوعی

مفهوم تأثیر موضوعی را با استفاده از یک مثال روشن می‌کنیم. فرض کنید احکام زیر، قسمتی از یک برنامۀ فرترن باشد:

(۱) P:=X+Y
(۲) Q:=Y/P
(۳) R:=X*P
(۴) S:=Q-R
(۵) T:=R*P
(۶) U:=S/T

در بخش دیگری از این برنامۀ فرترن، متغیرهای S، R، Q، P و T تنها به عنوان متغیرهای موقت مورد استفاده قرار گرفته‌اند. این متغیرهای موقت هیچگونه ارتباطی با متغیرهای مربوط به احکام (۱) تا (۶) در مثال مذکور ندارند.

منطق برنامۀ‌ها باید می‌کند که این دو قطعه از برنامۀ‌ها، بطور همزمان اجرا شوند، درحالی که به دلیل استفاده از اسامی واحد برای متغیرهای مختلف، این کار ممکن نیست (متأ سقانه بسیاری از زبانهای متعارف، این سبک برنامۀ‌سازی را تشویق می‌کنند). البته کامپایلر می‌تواند این مسأله را تشخیص دهد و از یک مجموعه متغیرهای موقت استفاده کند. ولی تشخیص این مسأله در زبان‌های مناسی مانند فرترن و پالان، ساده نیست. باید بررسی دقیقی برای تشخیص فعال یا غیرفعال بودن هر متغیر، صورت بگیرد. این بررسی زمانی دشوارتر می‌شود که استفاده از GOTO یا ساختارهای کنترل بی‌نظم دیگر نیز مجاز باشد. برای حل این مشکل، می‌توان برای هر متغیر، یک "حوزه" - ناحیه‌ای از برنامۀ‌ها که متغیر در آن، فعال است - تعریف کرد و دخول و خروج از بلوک‌های تشکیل دهنده آن حوزه را محدود ساخت. محدوده اثر هر متغیر، حوزه‌ای است که آن متغیر، در آن تعریف شده است.

اثرات جنبی^۹

اثرات جنبی به صورت‌های گوناگون ظاهر

زبانهای مبتنی بر جریان داده‌ها

ترجمه و تلخیص: میترا آل‌یاسین

مقدمه

برای رسیدن به هدف ایجاد همزمانی در اجرای دستورالعملها، تاکنون سیستمهای چندپردازنده‌ای، ماشینهای برداری و پردازنده‌های آرایه‌ای، طراحی و به کار گرفته شده‌اند. در مورد هر یک از این انواع معماری، تلاش‌های در زمینه طراحی کامپایلرهای که برنامۀ‌ها را نوشته شده به زبانهای متعارف را به صورت بهینه در می‌آورند، صورت گرفته است. به عنوان مثالی از این گونه کامپایلرها، می‌توان از کامپایلرهای برداری برای زبان فرترن نام برد. علاوه بر آن، زبانهای جدیدی نیز برای تسهیل استفاده از این گونه سیستمها طراحی شده‌اند که از آن میان می‌توان به پاسکال همزمان^۱ برای سیستمهای چندپردازنده‌ای، "گلیپر"^۲ برای پردازنده برداری "ایلیاک"^۳ (زبانیکه مستقیماً از خاصیت یک ماشین برداری استفاده می‌کند)، و گونه‌های برداری مختلف برای زبان فرترن، اشاره کرد.

این گونه زبانها، ویژگیهای برداری، آرایه‌ای یا چندپردازنده‌ای کامپیوتر را برای برنامۀ‌ها ساز، قابل رویت می‌سازند و به منزله وسیله‌ای هستند که برنامۀ‌ها را استفاده از آن، می‌توانند برای کشف احکامی که می‌توانند بطور همزمان اجرا شوند، به کامپایلر کمک کند. بسیاری از این زبانها غیرطبیعی هستند، به این مفهوم که به جای انعکاس نحوه تفکر برنامۀ‌سازان در حل مسائل، دقیقاً رفتار سیستمی را که برای آن طراحی شده‌اند، منعکس می‌سازند.

کامپیوترهای مبتنی بر جریان داده‌ها، همانند سایر انواع کامپیوترهای موازی، به زبانهای ویژه‌ای نیاز دارند. کارآئی اجرای برنامۀ‌های مبتنی بر جریان داده‌ها که در زبانهای متعارف مانند پالان و فرترن نوشته می‌شوند، بسیار پائین است. در حال حاضر، زبانهای مناسب برای کامپیوترهای مبتنی بر جریان داده‌ها، موجود هستند. ویژگی اساسی زبانیکه برای چنین کامپیوترهایی مناسب باشد، "اجتناب از اثرات جنبی" است. زبان لیسپ خالص، بهترین نمونه از زبانیکه است که در آن اثرات جنبی وجود ندارد.

در این مقاله، به زبان مبتنی بر جریان داده‌ها را مورد بررسی قرار خواهیم داد. دو زبان "وی ال"^۴ و "تی دی"^۵ به ترتیب در انستیتو

در این برنامه، چون A و B آرایه‌های متفاوت هستند، فراخوانی SORT2 با رجوع به B[J] می‌تواند به‌طور همزمان اجرا شوند، اما اگر این رویه، بخشی از یک برنامه بزرگتر بود و SORTREAD با حکم SORTREAD(Q,Q,M,N) فراخوانده می‌شد، آرایه‌های A و B یکسان می‌شدند. حتی اگر آرایه‌ها نیز استفاده نکنیم، در احکامی مانند احکام زیر دچار اشکال خواهیم شد:

(۱) A[J]:=3;

(۲) X:=A[K];

حکم (۱) باید آرایه A را به حکم (۲) بسپارد. اما در این صورت، حکمی مانند حکم انتسابی زیر به‌خود وابسته است:

for J=1 to 10 do

A[J+1]:=A[J]+1;

ولی اگر رکوردها و آرایه‌ها به‌وسیله احکام، دستکاری شوند به‌عنوان پارامترهای رویه‌ها رد و بدل شوند، غیرممکن است که بتوانیم بگوئیم چه موقع یک عنصر را به تغییر کرده است و این تغییر، چه اثراتی در یک قسمت دیگر برنامه می‌گذارد.

یک راه برای حل این مسأله، استفاده از "فراخوانی با مقدار" ۱۲ به‌جای روش متداول "فراخوانی با رجوع" ۱۳ است. در روش فراخوانی با مقدار، هر رویه، شناسه‌های خود را (حتی اگر آرایه‌ها باشند)، رونویسی می‌کند. به این ترتیب، هیچگاه نمی‌توانند شناسه‌ها حقیقی را در برنامه فراخواننده، تغییر دهد. علت استفاده از روش فراخوانی با رجوع این است که طبیعی‌ترین راه فکر کردن در باره محاسبات روی کامپیوترهای فون نویمانی است و ضمناً روی این گونه کامپیوترها، روش کار آتری است.

زبانهای اعمال شدنی ۱۴

در زبانهای مبتنی بر جریان داده‌ها، تمام آرایه‌ها "مقدار" هستند (به‌جای این که متغیر باشند). آرایه‌ها با احکام انتسابی نمایه‌دار، مانند حکم زیر تغییر نمی‌کنند:

A[J]:=S;

بلکه به‌وسیله عملگرهایی که آرایه‌های جدید ایجاد می‌کنند، مورد پردازش قرار می‌گیرند. ساده‌ترین عملگر برداری روی شناسه عمل می‌کند - یک آرایه، یک نمایه، و یک مقدار داده جدید. نتیجه عملیات، یک بردار جدید است که آن مقدار داده، در نمایه مشخص شده قرار دارد و مقدار دیرینه‌تر بردار در نمایه‌های دیگر، مانند بردار اولیه است.

در زبان "وی ای ال"، نحوه نوشتن این عملگر ابتدائی به‌صورت زیر:

A[J:S]

و در زبان "دی" به این صورت است:

A+[J]S

چنین حکمی، شناسه خود را تغییر نمی‌دهد. بنا بر این، در این برنامه وی ای ال:

(۱) B:=A[J:S];

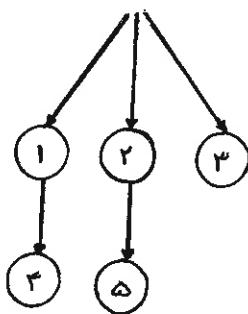
(۲) C:=A[K:T];

(۳) P:=A[L];

(۴) Q:=B[M];

(۵) R:=C[N];

احکام (۱) و (۲) با یکدیگر تداخل ندارند. ام از این که احکام (۱) و (۲) اجرا شده باشند یا نه، حکم (۳) می‌تواند بلافاصله اجرا شود، چون احکام مذکور هیچگونه اثری بر روی حکم (۳) ندارند. حکم (۴) به‌محض تکمیل اجرای حکم (۱) می‌تواند اجرا شود، اما از این که اجرای حکم (۲) به پایان رسیده باشد یا نه، ترتیب اجرای احکام در نمودار ۱ مشخص شده است.



نمودار ۱ - ترتیب اجرای احکام برنامه ۵

توجه کنید که این وضعیت، کاملاً مشابه وضعیتی است که در مثال ساده برنامه (۱) داشتیم. ترتیب اجرای دستورالعملها، کاملاً براساس وابستگی داده‌ها است و این دقیقاً منظور ما را از طراحی کامپیوترهای مبتنی بر جریان داده‌ها برآورده می‌سازد.

برنامه SORT2 در زبان وی ای ال (با حذف اعلانها) به این صورت نوشته می‌شود:

```

function SORT2(A,J)
  if A[J] < A[J+1] then A
  else (A[J:A[J+1]])[J+1:A[J]]
  end
end
  
```

توجه کنید که عملگر "ساختن آرایه"، می‌تواند با خود با عملگرهای دیگر ترکیب شود (درست مانند عملگرهای محاسباتی).

شاید در نظر گرفتن آرایه‌ها و رکوردها به صورت مقدار، عمیق‌ترین تفاوت بین برنامه نویسی در زبانهای مبتنی بر جریان داده‌ها و زبانهای متعارف باشد. دید معمول این است که آرایه‌ها در مکانهای ثابت حافظه قرار می‌گیرند و به‌وسیله احکامی که به ترتیبهای خاصی اجرا می‌گردند، دستکاری می‌شوند. چنانکه دیدیم، این نظریه با مفهوم اجرای موازی احکام، مغایرت دارد.

دید صحیح در مورد زبانهای مبتنی بر جریان

داده‌ها این است که آرایه‌ها و رکوردها را به صورت "مقدار" در نظر بگیریم. این "مقدارها" درست مانند مقادیر ساده (اعداد صحیح و اعداد حقیقی) به‌وسیله عملیات مختلف، دستکاری می‌شوند. همچنانکه یک عدد صحیح را می‌توان به صورت رشته‌ای از ارقام در نظر گرفت، یک آرایه صحیح را نیز می‌توان به‌صورت رشته‌ای از اعداد صحیح در نظر گرفت. اگر مقدار J، ۳۹۴۱۶ باشد، حکم K:=J-400 باعث می‌شود که در متغیر K، مقدار ۳۱۰۱۶ قرار بگیرد. هیچ برنامه‌ساز انتظار ندارد که مقدار J هم تغییر کند. اکنون اگر A یک آرایه با عناصر (۳، ۴، ۵، ۶) باشد، حکم:

B:=A[3:0];

کاملاً مشابه حکم قبلی است. عناصر B عبارت خواهند بود از: (۳، ۴، ۵، ۶) و مقدار A نیز به هیچوجه تغییر نمی‌کند.

زبانهای کدهای آنها انجام تمام پردازشها، از طریق عملگرهایی که به "مقادیر" اعمال می‌شوند صورت می‌گیرد. "زبانهای اعمال شدنی" نامیده می‌شوند و به همین دلیل، زبان طبیعی برای محاسبات مبتنی بر جریان داده‌ها هستند. اولین زبان کاربردی که روی یک کامپیوتر پیاده شده است، زبان لیسپ است (البته در صورتی که RPLACA و RPLACD بقیه توابقی را که دارای اشارات جنبی هستند از آن حذف کنیم که در این صورت به آن "لیسپ خالص" گفته می‌شود).

بررسی کارآئی زبانهای اعمال شدنی

تاکنون مباحثات زیادی بر سر کارآئی زبانهای اعمال شدنی صورت گرفته است. ادعا می‌شود که کارآئی اجرای بسیاری از الگوریتمها در زبانهای متعارف، بسیار بیشتر از زبانهای اعمال شدنی است. علت این ادعا این است که هر برنامه نوشته شده برای یک کامپیوتر متعارف "فون نویمانی" را می‌توان با در نظر گرفتن کل فضای حافظه به صورت یک آرایه، در یک زبان اعمال شدنی باز نویسی کرد. این آرایه از یک عملیات به عملیات دیگر منتقل می‌شود. بنا بر این، اجرای عملیات در چنین سیستم اعمال شدنی کاملاً ترتیبی است و هیچ همزمانی نمی‌تواند وجود داشته باشد. اما یک برنامه‌ساز ما هر، می‌تواند در اجرای احکام نوشته شده به یک زبان متعارف، همزمانی ایجاد کند و سبب شود که اجرای برنامه، کارآئی بیشتری از سیستم اعمال شدنی داشته باشد. به برنامه متعارف زیر توجه کنید:

(۱) A[J]:=S;

(۲) A[K]:=T;

(۳) P:=A[M];

(۴) Q:=A[N];

اگر برنامه‌ساز بداند که مجموعه نمایه‌های آرایه A را می‌تواند به دو مجموعه مجزا تقسیم ورق بزنید

اشیاء صحت برنا مهنا مناسب باشند، امری ضروری است چون ادعای $J=1$ کاملاً بی‌معنی است.

درحقیقت، هیچ الزامی وجود ندارد که یک زبان مبتنی برجریان داده‌ها از قانون انتساب جریان داده‌ها با انتسابات چندگانه طراحی کرد که در آن، حوزه هر متغیر از یک تعریف تا تعریف بعدی آن متغیر، گسترش یابد. درواقع، تعریف بعدی، یک متغیر جدید ایجاد می‌کند که بر حسب اتفاق همان نام متغیر قبلی را دارد. برنا مه‌ای را که به این صورت نوشته شده باشد، می‌توان به سادگی به برنا مه‌ای که از قانون انتساب یکا نه تبعیت می‌کند تبدیل کرد:

برای هر متغیری که دوباره تعریف کرده‌اید یک نام جدید انتخاب کنید و در هر جای برنا مه که به متغیر قبلی ارجاع کرده بودید، به این نام جدید ارجاع کنید.

حلقه‌های تکرار

اکنون این سوال پیش می‌آید که در رابطه با احکامی مانند $I:=I-1$ یا $A:=A[J:X+Y]$ چه می‌توان کرد؟ درحلقه‌های تکرار که وجودشان در برنا مه‌ها ضروری است، دیگر نمی‌توان از قانون انتساب یکا نه استفاده کرد. تکنیک تعویض نام متغیرها، برای ایجاد برنا مه‌ای که از قانون انتساب یکا نه تبعیت کند، تنها در برنا مه‌هایی که از زبان پایه‌ها شین، به ترتیب اجرا می‌شوند، مؤثر است. اگر در برنا مه‌ای، حلقه تکرار وجود داشته باشد، این تکنیک منظر برنا مه‌ها را برآورد نمی‌آورد. به این مثال توجه کنید:

```
for I:=1 to 10 do
  J:=J+1;
```

حکم فوق الذکر را نمی‌توان به این حکم تبدیل کرد:

```
for I:=1 to 10 do
  J1:=J+1;
```

وجود GOTO این مشکل را وخیم‌تر می‌کند. اما زبان‌های مبتنی برجریان داده‌ها، فاقد حکم GOTO هستند و نیاز به حلقه‌هایی دارند که با ساختارهایی مانند WHILE... DO... و احکام مشابه موجود در پیل وان و پاسکال ایجاد می‌شوند. این موضوع سبب می‌گردد که حل این مشکل، آسان شود. برای ایجاد معادلی برای ساخت WHILE... DO... در این گونه زبان‌ها، باید تشخیص بدهیم که بخش "DO" از چن‌ین ساختی شامل چه قسمت‌هایی است. بطور کلی، هر حلقه تکرار مشتمل است بر:

- ۱- تعریف مقدار اولیه متغیرهای حلقه
- ۲- گزاره‌ای که مشخص می‌کند آیا برای هر چرخه معین از متغیرهای حلقه، حلقه باید ادامه پیدا کند یا ختمه یابد
- ۳- اگر حلقه‌ها بدینا تمه پیدا کند، یک عبارت، مقدار (مقادیر) را که باید به برنا مه بازگردانده شود، مشخص می‌آورد. این مقادیر به مقدارهای فعلی متغیرهای حلقه بستگی دارند

عملیاتی که از S استفاده می‌کنند، صورت بگیرد. بنا براین به حکم انتسابی $S:=X+Y$ می‌توان به صورت یک "تعریف" نگاه کرد. زبان‌هایی که احکام انتسابی را به این صورت تفسیر می‌کنند، "زبان‌های تعریفی" نامیده می‌شوند. چنین زبان‌هایی برای اشیاء صحت برنا مه‌ها بسیار مناسب هستند زیرا ادعاهای 16 که با پید برای اشیاء صحت برنا مه‌ها فاش شوند، دقیقاً همان تعاریفی هستند که صراحتاً در برنا مه‌ها هر گشته‌اند. در زبان‌های متعارف، ممکن است متغیرهای S و X و Y بارها در حین اجرای برنا مه تغییر کنند. به همین دلیل برای تعیین این که ادعاهای ما مانند $S=X+Y$ در چه قسمت‌هایی از متن برنا مه درست هستند، باید جریان کنترل برنا مه، دنبال شود. بنا بر این، ادعا‌ها باید به دقت طی در برنا مه، وابسته شوند. در یک زبان تعریفی، اگر یک بلوک از برنا مه، حاوی حکم $S=X+Y$ باشد، ادعای $S=X+Y$ در آن بلوک درست است. البته باید دقت شود که تعاریف گردشی مانند زیر صورت نگیرد:

$$X:=Y;$$

$$Y:=X+3;$$

می‌توانیم کامپایلری داشته باشیم که هر نوع تعریفی را در صورتی که یک "ترتیب سازگار" را رعایت کند، مجاز باشد. منظور از یک ترتیب سازگار، ترتیبی است که در آن، ارجاع به هیچ نامی (در سمت راست یک تعریف) پیش از این که آن نام قبلاً تعریف شده باشد، صورت نمی‌گیرد. این شرط به سادگی به وسیله کامپایلر، قابل بررسی است. بنا براین، رابطه $S=X+Y$ در تمام احکامی که پس از تعریف S ظاهر شده‌اند، درست است. اما از آنجا که S در هیچ یک از احکام پیش از تعریف S ظاهر نشده است، می‌توان ادعا کرد که رابطه مذکور در تمام بلوک‌های حوزه S را شامل می‌شود، درست است.

قدرت زبان‌های تعریفی در اشیاء صحت برنا مه‌ها به خوبی به‌شود رسیده است. "لوسید" یک نمونه از زبان‌های کاربردی تعریفی است که به منظور تسهیل اشیاء صحت برنا مه‌ها طراحی شده است.

یکی از مشکلاتی که زبان‌های تعریفی با آن روبرو هستند، "تعریف چندگانه یک نام واحد" است. گفتیم زبان‌های تعریفی از قانون انتساب یکا نه تبعیت می‌کنند: یک نام در حوزه خود فقط یکبار می‌تواند در سمت چپ یک حکم انتسابی (تعریف) ظاهر شود. قانون انتساب یکا نه، چنین انتساب‌هایی را مجاز نمی‌داند:

$$J:=J+1;$$

چون ظهور J در سمت راست حکم فوق پیش از تعریف J صورت گرفته است، یک ترتیب نام‌سازگار از احکام را ایجاد می‌کند و کامپایلر، آن را به عنوان یک "خطا" به شما می‌آورد. البته ممنوعیت چنین تعاریفی در زبان‌هایی که برای

کند، بطوری که J و M در یک مجموعه و K و N در مجموعه دیگری باشند، احکام (۱) و (۲) می‌توانند بطور همزمان اجرا شوند، (۲) بلافاصله پس از (۱) می‌تواند اجرا شود. (۴) فقط نیاز دارد که (۲) پیش از آن اجرا شده باشد. چنانچه برنا مه‌ها، توانائی کنترل همزمانی بطور صریح را نداشته باشند (مثلاً به وسیله "سما فورها"، می‌توانند دقیقاً این محدودیتها را مشخص سازد. اما این عمل، بدون تعبیه یک مکانیزم اضافی در زبان‌های مبتنی بر جریان داده‌ها، امکان پذیر نیست. البته برای رفع این اشکال، می‌توان آرایه‌ها به دو بخش مجزا تقسیم کرد. هر بخش را جداگانه مورد پردازش قرار داد و سپس در موقع لزوم، دوبخش را با یکدیگر ترکیب کرد. به عنوان مثال، فرض کنید آرایه A1 تنها شامل عناصر J و M و آرایه A2 تنها شامل عناصر K و N باشد. در این صورت، احکام زیر می‌توانند منظور ما را برای ایجاد همزمانی در اجرای عملیات برآورده سازند:

$$B1:=A1[J:S];$$

$$B2:=A2[K:T];$$

$$P:=B1[M];$$

$$Q:=B2[N];$$

البته راه‌های دیگری نیز برای غلبه بر این مشکل که آرایه‌ها همزمانی اجرای عملیات را محدود می‌کنند، وجود دارد که بعداً به یکی از آنها اشاره خواهیم کرد.

زبان‌های تعریفی ۱۵ و قانون انتساب یکا نه

به استثنای حلقه‌های تکرار که بعداً مورد بحث قرار خواهند گرفت، تنها کاری که یک حکم انتسابی می‌کند، این است که پس از تعیین یک مقدار، آن را به نامی که در سمت چپ حکم قرار دارد، نسبت می‌دهد. نتیجه انتساب، تنها در عباراتی که پس از آن نام ظاهر شده‌اند، قابل استفاده است. چنانچه عبارت سمت راست یک حکم انتسابی، با متغیر سمت چپ حکم، جایگزین گردد، برنا مه حاصل با برنا مه اولیه، معادل خواهد بود. دو قطعه برنا مه (A) و (۹) معادلند.

$$S:=X+Y;$$

$$D:=3*S;$$

$$E:=S/2+F(S);$$

(۸)

$$D:=3*(X+Y);$$

(۹)

$$E:=(X+Y)/2+F(X+Y);$$

واضح است که قطعه برنا مه (۸) کارآمدتر است چون تنها به دو جمع - به جای چهار عمل جمع در قطعه برنا مه (۹) - نیاز دارد.

حکم $S:=X+Y$ دقیقاً همان مفهوم معادله ریاضی $S=X+Y$ را می‌رساند. یعنی در حوزه متغیرهای مذکور، S حاصل جمع X و Y است. این تطابق، همیشه وجود دارد. البته اضافه کردن X و Y و تشکیل S باید پیش از انجام

جایی که به یکدیگر وابستگی نداشته باشند)، اجرا شوند.

تنگنای دیگری که در محاسبات مبتنی بر جریان داده‌ها وجود دارد، لزوم محاسبه تمام عناصر یک آرایه، پیش از دستیابی به هر یک از عناصر آن آرایه می‌باشد. اگر تابع F هر بار یک عنصر از آرایه را ایجا دکنند سپس به تابع G که آنهم هر بار یک عنصر از آرایه مذکور را می‌خواند منتقل سازد، اجرای G تنها پس از اتمام F می‌تواند شروع شود. در بسیاری از این موارد، چنین تابه‌گیری ضروری نیست. روشهای متفاوتی برای از بین بردن این تابه‌گیری وجود دارد. یک روش که قبلاً به آن اشاره شد، تقسیم آرایه به چند قطعه با استفاده از قلام داده جداگانه به جای یک آرایه است. این روش کاملاً عمومیست و در طولی لازم است که برنامهمسا زمان محاسبه کند که در هر لحظه، کدام قسمت از آرایه مورد نیاز است. راه دیگر غلبه بر تنگنای آرایه‌ها، استفاده از جریانها^{۲۰} است. یک جریان، آرایه‌ای است که در طول زمان، تکه تکه می‌شود و در هر لحظه، یک عنصر آن مورد پردازش قرار می‌گیرد. در مثال قبلی که F هر بار یک عنصر از آرایه را ایجا می‌کرد و سپس آرایه را به G منتقل می‌ساخت، راه طبیعی برای ایجا دهمزمانی، استفاده از یک جریانست. به محض این که F یک عنصر از آرایه را ایجا کرد، G آن را دریافت می‌کند. به این ترتیب هنگامی که F عنصر $I+1$ م آرایه را محاسبه می‌کند، G عنصر I م را مورد پردازش قرار می‌دهد و نتیجتاً محاسبات F و G به صورت موازی انجام می‌شود. خواننده علاقه‌مند به کسب اطلاعات بیشتر در این زمینه، می‌تواند به مرجع این مقاله مراجعه کند.

پاداشتها و مرجع

- * Ackeman W.B. "Data Flow Languages" AFIPS Conf. Proc. vol 48 (1979 National Computer Conf.) AFIPS press, Montvale N.J. pp 1087-1095
- 1: Concurrent Pascal
 - 2: GLYPNIR
 - 3: Illiac IV
 - 4: VAL
 - 5: ID
 - 6: Massachusetts Institute of Technology
 - 7: University of California at Irvine
 - 8: LUCID
 - 9: locality of effect
 - 10: side effects
 - 11: Identifiers
 - 12: call by value
 - 13: call by reference
 - 14: applicative
 - 15: definitinal
 - 16: assertions
 - 17: overflow
 - 18: manipulation of global status flags
 - 19: underflow
 - 20: streams

```
initial I ← N; J ← 1;
while I ≠ 0 do
  new I ← I-1;
  new J ← J*I;
return J;
(۱۱) حلقه تکرار برای محاسبه فاکتوریل N (در زبان آی دی)
```

ایجا دپخش خط نگهداری کند، آن رکورد، رد شرح خط را (خط چه موقع و در کجا اتفاق افتاده است و کدام حلقه‌ها و رویه‌ها در آن زمان، فعال بوده‌اند) مشخص می‌سازد. اگر هدف این باشد که در صورت بروز خطا، آن خطا تصحیح گردد. احتمالاً می‌توان لیستی از خطاهای رادریسک آرایه نگهداری کرد. این کار را می‌توان با استفاده از عملیاتی که خطاهای آرایه‌ها را می‌سازد، انجام داد. به عنوان مثال، احکام زیر بر حسب وقوع یا عدم وقوع خطا، در متغیر Z ، مقدار صفر یا X/Y را قرار می‌دهند. این احکام در زبان "وی ای ال" نوشته شده‌اند:

```
ZZ:=X/Y;
Z:=if is-error(ZZ) then 0 else ZZ end;
```

روشهای حصول حداکثر همزمانی

برای حصول حداکثر همزمانی، لازم است که محاسبات تازمانی که الزامی در کار نباشد، بطور ترتیبی اجرا نشوند. ساختهای تکراری که قبلاً مورد بررسی قرار دادیم، اجرای ترتیبی چرخه‌های متفاوت را ایجا می‌کنند. اگر مقادیر متغیرهای حلقه در یک چرخه، وابسته به مقادیر آنها در چرخه پیشین باشد (چنان که در محاسبه فاکتوریل بود)، برای ایجاد همزمانی هیچکاری نمی‌توان کرد. البته یک کامپیوتر مبتنی بر جریان داده‌ها، اغلب می‌تواند قسمتی از یک چرخه را پیش از اتمام چرخه قبلی اجرا کند. چنانچه مقادیر در یک چرخه، وابسته به چرخه‌های پیشین نباشد، چرخه‌ها می‌توانند به صورت موازی اجرا شوند. در زبان "وی ای ال"، این کار از طریق ساختی به نام "for all" انجام می‌شود. استفاده از این ساخت، تنها در صورتی مجاز است که هیچ چرخه‌ای به چرخه دیگر ارتباط نداشته باشد. در این صورت، تمام چرخه‌ها به صورت موازی اجرا خواهند شد. در زبان "آی دی"، همین کار از طریق علامت زدن خودکار متغیرهای تکراری شماره چرخه‌ها، انجام می‌شود. چرخه‌ها می‌توانند خارج از ترتیب یا بطور همزمان (در هر

حکام

```
for I,J:=N,1;

do if I=0
then J
else iter I,J:=I-1,J*I
```

(۱۰) حلقه تکرار برای محاسبه فاکتوریل N (در زبان وی ای ال)

۴ - اگر حلقه‌ها پیدا داده‌ها بد، عباراتسی مقادیر جدید متغیرهای حلقه را مشخص می‌کنند. این مقادیر به مقادیر فعلی متغیرهای حلقه بستگی دارند.

در برنامهای (۱۰) و (۱۱)، حلقه تکراری را که فاکتوریل N را محاسبه می‌کند (با حذف اعلانها) در زبانهای "وی ای ال" و "آی دی" مشاهده می‌کنید. نمایش برنامه در "لوسید" نیز مشابه "آی دی" است.

با این که مقادیر متغیرهای حلقه تغییر پیدا می‌کنند، این تغییر فقط بین یک چرخه تکرار چرخه بعدی صورت می‌گیرد. قانون انتساب $I:=I+1$ نشان می‌دهد که از زمان ریفری ما نند $I:=I+1$ ما نعت به عمل می‌آورد، همچنان در درون هر چرخه، مستمراست. این قانون، تنها در مرز بین چرخه‌ها، اعتبار خود را از دست می‌دهد. در زبان "وی ای ال"، تعریف مجدد یک متغیر، تنها پس از کلمه `iter` (که دستوری برای آغاز یک چرخه جدید از حلقه تکرار است) مجاز می‌باشد. در زبانهای "آی دی" و "لوسید"، در مرز بین هر دو چرخه، مقدار `new` به مقادیر فعلی چرخه جدید تبدیل می‌گردد. از آنجا که در هر چرخه از حلقه تکرار، از قانون انتساب یگانگی تبعیت شده است، ادعاهای به صورت زیر در می‌آیند:

$S=X+Y$

در هر چرخه

خطاهای ارتباط استثنائی

لازمه تابه‌شیرموضعی این است که در کنترل خطاهای ما نند سرریز محاسبات^{۱۷}، به جای ایجا وقفه در برنامه‌ها "دستکاری شاخصی وضعیت سرریز"^{۱۸}، از "مقادیر خطا" استفاده شود. چنانچه حین اجرای عملیاتی، خطا پیش آید، این خطا تنها باید به مقصدهای آن عملیات منتقل شود. برای این منظور، می‌توان مجموعه مقادیر را بزرگتر کرد بطوری که مقادیر خطا ما نند سرریز، پاریز^{۱۹}، و تقسیم بر صفر را نیز در برگیرد. چنانچه بی‌نتیجه گذاشتن محاسبات در هنگام وقوع خطا مورد نظر باشد، این هدف به سادگی با پخش کردن مقادیر خطا، قابل حصول خواهد بود. وقتی که خطا تا انتهای بدنه یک حلقه پخش شود، به جای این که چرخه بعدی حلقه شروع شود، اجرای حلقه برای همیشه ادامه می‌یابد. به این ترتیب، کل محاسبات، سریعاً خاتمه می‌یابد و یک مقدار خطا را به عنوان نتیجه، باز می‌گرداند. چنانچه کامپیوتر رکوردی از هر

توضیح

مقادیر اولیه N و 1 به ترتیب در متغیرهای حلقه I و J قرار می‌گیرند. به سمت پایین می‌شمارد. J حال ضرب I و J را نگه می‌دارد.

پایان حلقه؟
آری، نتیجه نهایی در J قرار دارد.
نه. مقادیر I و J جدید محاسبه می‌شوند و چرخه جدید آغاز می‌گردد.



مفحه نمایش رنگی و یک سیستم کوچک تحلیل -
کننده اشعه ایکس نیز می باشد. قابلیت
تفکیک پذیری ۳ مفحه نمایش مزبور که تنها ویر
حاصل را در معرض دید قرار می دهد،
۵۱۲x۵۱۲ نقطه بوده و کیفیت بالایی (بدون
لرزش) برخوردار است.

به نقل از

Electronic Product News

May 1985

- 1: Scanning Microscope
- 2: Carl Zeiss
- 3: Resolution

شرکت امدال^۱ و ساخت آمریکا^۲ میکروتر

شرکت آمریکایی امدال که اولین سازنده
کامپیوترهای بزرگ^۳ هم از آی بی ام محسوب
می شود، اکنون در نظر دارد تا آمریکا میکروتری
هم از آی بی ام به وجود آورد. هدف شرکت
مزبور، ایجاد کامپیوتر قدرتمندی است که بهتر
از دیگر کامپیوترهای بزرگ موجود، در زمینه
های گوناگون علمی نظیر هواشناسی و نیـ
ز کاربردهای همه منظوره گشامانجام معاسبات

دا ده است اما قبل از بازاریابی گسترده تر، در
انتظار تعیین استانداردهای رسمی قرار دادن
آنها در اختیار تولیدکنندگان تجهیزات اصلی
می باشد. قیمت مورد نظر سونی، چیزی در حدود
۲۰٪ افزون بر بهای دیسکهای ۱ مگابایتی،
پیش بینی شده است.

در همین حال شرکت توشیبا نیز در شرف
عرضه دیسکهای ۳ ۱/۲ اینچی با ظرفیت ۴ مگا-
بایت است که جنس پوشش مغناطیسی آنها با
"فرات با ریم" می باشد. اولین نمونه های تحت
آزمایش این گونه دیسکها در اواخر سال آینده
ارائه خواهد گردید. پیش بینی می شود که در سال
۱۹۸۷، سری اول این دیسکها به معرض فروش
گذاشته شود.

به نقل از

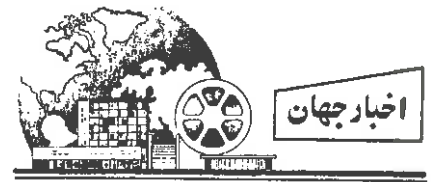
BYTE

OCT. 1985

- 1: Floppy Disks
- 2: Formatting
- 3: Track
- 4: Drives
- 5: OEM (Original Equipment Manufacturer)

میکروسکوپ پوشش^۱

شرکت کارل زایس^۲ جدیداً "میکروسکوپ
الکترونی پوشنده ای به نام DSM950 به بازار
عرضه داشته که قادر است کلیه عملیات الکترو-
نی - نوری خود را به صورت رقمی کنترل نموده
و علائم تصویری حاصل را به صورت رقمی پردازش
نماید. پیکربندی این میکروسکوپ شامل یک



چاپ رنگی با کیفیت و قیمت بالا^۱

در نما یسگا هگرافیک کامپیوتری به نام
"سیگ گراف" که در سانفرانسیسکو برگزار شد،
شرکتهای متعددی چاپگرهای خود را که قابلیت
چاپ رنگی با قدرت تفکیک پذیری ۲ بالا برخوردار
دار بودند، ارائه نمودند. بطور مثال، چاپگر
مدل CGP-400 شرکت هیتاچی با قدرت چاپ ۴۰۹۶
رنگ و قابلیت تفکیک پذیری ۴۰۰ نقطه در اینچ
به قیمت ۱۰،۰۰۰ دلار عرضه شد. شرکت پاناسو-
نیک نیز چاپگر مشابهی را به معرض نمایش
گذاشته بود.

نمونه چشمگیر دیگر، متعلق به شرکت

فوجی (تولیدکننده فیلمهای عکاسی) بود که
توسط شرکت میتسوبشی توزیع می گردد. این
چاپگر قدرت چاپ پوستر را بعد از بزرگ (تسا
۲۳x۲۸ اینچ مربع) را ظرف مدتی کمتر از ۱۵
دقیقه داشته و قابلیت تفکیک پذیری آن ۱۵۲
نقطه در اینچ می باشد. این چاپگر می تواند تا
۲۶۲۰۰۰ رنگ را چاپ نماید. قیمت چاپگر مزبور
یکمدهزار دلار می باشد.

به نقل از

BYTE

OCT. 1985

- 1: SIGGRAPH
- 2: Resolution

سونی و ساخت دیسکهای لرزان^۱ ۳ ۱/۲ اینچی

شرکت سونی که پیش از این دیسکهای لرزان
۳ ۱/۲ اینچی با ظرفیت ۱ مگابایت را به بازار
عرضه داشته، اخیراً "دروکیوا" علام نموده که
موفق به ساخت دیسک لرزان ۳ ۱/۲ اینچی با
ظرفیت ۲ مگابایت گردیده است که پس از قالب
بندی می تواند تا ۱/۶ مگابایت اطلاعات را
ذخیره نماید. تعداد قطعات در هر شیاف^۲ این
دیسک معادل دو برابر تعداد معمول بوده و
فضا مت پوشش مغناطیسی آن ۱ میکرون می باشد
(نصف فضای پوشش دیسک ۱ مگابایتی قبلی).
با وجودی که تکنولوژی بکار رفته در ساخت
دیسک ۱ مگابایتی قدیم و دیسک ۲ مگابایتی
جدید نیز گرداننده های آنها مشابه هستند،
لیکن نوک مغناطیسی گرداننده های دیسکهای
جدید به میزان دقت بیشتری نیازمند می باشد.
ضمناً جهت تشخیص گرداننده نسبت به ۱ یا ۲
مگابایتی بودن دیسک، یک سوراخ اضافی بر
روی دیسک تعبیه گردیده است.
اگرچه تاکنون شرکت سونی به ۳۸ شرکت
سازنده مجوز ساخت دیسکهای ۲ مگابایتی را

هند در صد توسعه تکنولوژی ساخت کامپیوتر

آن دسته از شرکتهای کامپیوتری که با رکود
نسبی بازار کامپیوتر مواجه شده اند می خواهند
برخی از کارخانه های خود را تعطیل کنند، چشم
امید به هند دوخته اند که در صد و در کردن تکنو-
لوژی ساخت کامپیوتر به منظور افزایش تولید
داخلی خود است.

تاکنون، هند به دلیل در اختیار داشتن
نیروی انسانی و فروکار^۳ مدد رفته نرم افزار،
بیشتر به این جنبه از مورکا میکروتری توجه
داشته است، اما بررسیهای اخیر دولت آن
کشور (به رهبری راجیو گاندی، که شخصاً توجه
زیادی به تکنولوژی کامپیوتر دارد) نشان
داده اند که موقعیت مساعدی برای فعالیت
بیشتر در جهت تقویت تکنولوژی ساخت کامپیوتر
به وجود آمده است.

درواقع هند به صورتی زیرکانه از همزمانی
دور ویداد (افزایش نیازهای داخلی و رکود
نسبی بازار جهانی) استفاده کرده و می خواهد
تکنولوژی مورد نیاز خود را با پرداخت کسری از
قیمت یک سال پیش آن تأمین نماید. به این
منظور، هیأت خریدی برای شروع کار به زان

اعزام شده است.

بطور سنتی، سیاست هند چنین بوده است که
بخش مهمی از نیازهای سخت افزاری خود را در
داخل کشور تأمین نماید، حتی اگر مجبور باشد
در بدهد. این کار را با کمک شرکتهای بزرگ چند
ملیتی انجام دهد. بنابراین، وارد کردن
کارخانه های کامل و نصب آنها در هند تا حدودی
با سیاست سالهای قبل آن کشور مغایر است، اما
اخیراً عین همین کار در مورد خرید یک کارخانه
کامل از ایالتی برای ساخت موتورسیکلت
"لامبرتا" نیز انجام شده است.

نمونه بارزی از توانایی هند در جهت
فعالیت مستقل اقتصادی، صنعت خودروسازی آن
کشور است که در حال حاضر ۹۹٪ نیازهای داخلی
هند را تأمین میکند.

چنین به نظر میرسد که هند بیشتر به دنبال
خرید از شرکتهای است که به دلیل عدم امکان
رقابت با غولهای چینی آی بی ام و آپل، در
فکر خارج شدن از بازار کامپیوتر با تغییر دادن
جهت فعالیتها می خود هستند.

ترجمه آزاد: بهروز پیرامی
از: کامپیوتینگ، اکتبر ۱۹۸۵

وظایف این دستگاه است. عیب یاب مزبور به صفحه تلویزیون کوچکی مجهز است که حاصل محاسبات و اطلاعات بروز در آن مده، بر روی آن نمایش داده می شود. اپراتور می تواند دستور-العملهای دستگاه را از قبیل درون حافظه آن قرا ردهد.

به نقل از

Electronic Product News
May 1985

اخبار کوتاه

— علیرغم توافق شرکت هولیس برادرز مبنی بر خرید قسمت اعظم شرکت سینکرونیت آن از ورطه ورشستگی، این شرکت به دلیل آن که ارزش موجودی انبارهای سینکرونیت از مقدار واقعی ادعا گردیده بود، توافقنامه مزبور را ملغی اعلام داشت.

— دانشجویان دانشگاه استنفورد اخیراً موفق به ساخت یک دست مکانیکی شده اند که می تواند به زبان اشاره با افرادی که همگروه کسور هستند، "صحبت" نماید. دست مزبور نسبت به متون وارد شده از طریق پایانه های کامپیوتر عکس العمل نشان می دهد.

— شرکت AT&T به همراه شرکت نرم افزاری یونی سافت ساخت فاضلی را برای زبان ژاپنی تحت سیستم عامل یونیکس برای کامپیوترهای AT&T مدل 3B اعلام داشته است.

به نقل از

BYTE

OCT. 1985

1: Hollis Brothers

2: Unisoft

3: Interface

قرارداد همکاری میان اپل و ونگ

اخیراً شرکت اپل که نگران وضعیت بازار کامپیوترهای شخصی تجاری خویش است، پس از بررسیهای متعدد، با شرکت ونگ که از اعتبار چشمگیری برخوردار می باشد، یک قرارداد همکاری را آغاز نموده است. حیطه این مشارکت وسیع بوده و عمدتاً در زمینه های بازاریابی کامپیوترهای شخصی و نیز گسترش تولیدات، متمرکز خواهد بود. پیش بینی می شود که برقراری این ارتباط برای هر دو شرکت موفقیت آمیز و پر منفعت باشد. نظر شرکت اپل، بهره برداری از امکانات و توفیقاتی است که شرکت ونگ در بازار خودکار سازی دفاتر بدست آورده است. این شرکت بر آن است که در آینده در زمینه های ایجاد شبکه های محلی، ساخت چاپگرهای لیزری و نیز ایجاد دفاترهای لازم (نرم افزاری و سخت افزاری) برای هماهنگ نمودن ماشینها و شبکه های کامپیوتری اپل با محصولات آی بی ام فعالیت بیشتری داشته باشد.

به نقل از

Financial Times
Jan. 1985

1: Apple

2: Wang

3: Office Automation

4: Local Area Network (LAN)

5: Interfaces

عیب یاب خطوط تلفنی

اخیراً شرکت آلفا نای CAF دستگاه عیب یابی ساخته است که به وسیله ریزپردازنده کنترل شده و دارای ۶۴ کیلوبایت حافظه می باشد. تشخیص نوع عیب و محل آن در خطوط ارتباط تلفنی و نیز زمان سبب طول خطوط از جمله

رباضی در حجمی زیاد می باشد، بکار گرفته شده است. بدیهی است که بکار کامپیوترها در انجام این گونه وظایف سنگین، بهره دمی بالاتری را ارائه می دهند.

طبق ادعای شرکت امدال، اپرکا کامپیوتر مزبور با اپرکا کامپیوترهای سری ۴ به راحتی قابل رقابت خواهد بود. در حال حاضر، تنها در حدود ۱۰۰ اپرکا کامپیوتر در سرتاسر دنیا نصب گردیده و در حال فعالیت هستند. این رقم در مورد کامپیوترهای بزرگ بیش از ۵۰۰۰ است که اکثر آنها با آی بی ام هستند و با همسازان آن، اپرکا کامپیوتری که شرکت امدال عرضه خواهد داشت، مشابهمندی زیادی خواهد داشت. فوجیتسو (سهمدار اصلی شرکت امدال) در اوایل سال ۱۹۸۴ ارائه گردید. قیمت اپرکا کامپیوتر امدال بسته به اندازه آن بین ۷/۷ تا ۲۰ میلیون دلار برآورد شده است که به ازای فروش هر اپرکا کامپیوتر مبلغی معادل ۱ میلیون دلار برای این شرکت منفعبت به همراه خواهد داشت. طبق پیش بینی شرکت مزبور، ۶ اپرکا کامپیوتر در سال ۱۹۸۵ فروخته خواهند شد. مسئولین امدال معتقدند که علیرغم بهای سنگین این ماشینها، سرعت بالایی که ارائه می دهند خریداران را برای استفاده کنندگان از عمده نامه ها و دانشگاهها، مراکز دولتی، شرکت های نفتی و غیره مقرون به صرفه می نماید.

شرکت امدال در سال ۱۹۷۰ توسط جیمز امدال بنیان گذاری شد و از سال ۱۹۸۰ محور اصلی فعالیت های تولیدی این شرکت، همسازی با محصولات آی بی ام قرار گرفت. موفقیت امدال، همگان و به خصوص آی بی ام را متعجب ساخت و چنان که این شرکت از آن تاریخ تا کنون به عنوان بزرگترین تولیدکننده کامپیوترهای بزرگ همساز آی بی ام که ۸۰٪ فروش کامپیوترهای بزرگ در دنیا را در اختیار دارد، در بازار کامپیوتری مانده است.

سود این شرکت در سال ۱۹۸۳ معادل ۴۶ میلیون دلار (۶ برابر سال ۱۹۸۲) بوده که از مجموع ۷۸۸ میلیون دلار فروش آن بدست آمده است. علیرغم موفقیت های فوق، امدال با مشکل عمده ای مواجه است و آن خریداری شدن بخش زیادی از سهامش به وسیله ژاپنیها است. شرکت فوجیتسو در اوایل سال جاری میزان سهام خود را به ۴۹/۵٪ رساند و بر اثر همین ازدیاد نفوذ و قدرت ژاپنیها، جیمز امدال شرکت مزبور را ترک کرد. روابط فوجیتسو و امدال، تا حدی نزدیک شده است که طی موافقتنامه ای قرار شد امور مربوط به تحقیق و توسعه کامپیوترهای بزرگ از این به بعد توسط هر دو شرکت مشترکاً انجام پذیرد.

به نقل از

Economist

Oct. 1984

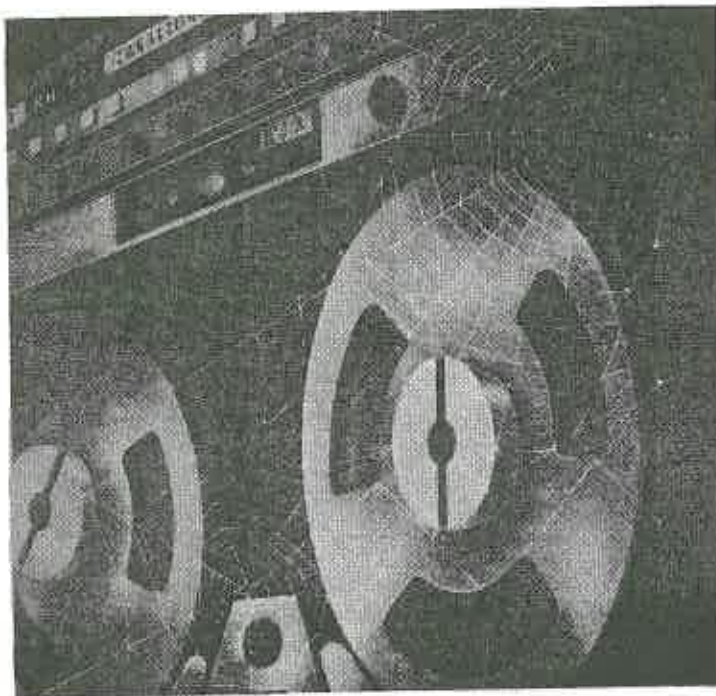
1. Amdahl

2. Supercomputer

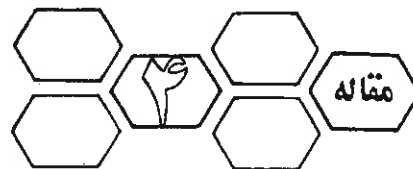
3. Mainframes

4. Cray

5. Gene Amdahl



مقاله



مروری بر صنعت داده پردازی اروپا در سال ۱۹۸۴

ترجمه و تالیف: لیلی حاجی اسماعیلی

شرکت‌های کامپیوتری اروپائی و نیز شعب اروپائی شرکت‌های آمریکائی، سال ۸۴ را سال پرباری یافتند. درآمد این گونه شرکت‌ها در این سال جمعا به ۲۸/۴ میلیارد دلار رسید که نسبت به سال قبل، ۹/۷٪ رشد نشان می‌دهد. از میان کلیه شرکت‌های فوق، ۲۵ شرکت که بالاترین میزان درآمد را داشته‌اند در جدول (۱) دیده می‌شوند. از این میان، ۱۲ رتبه متعلق به شرکت‌هایی است که مقر اصلی آنها، آمریکا می‌باشد. تعداد شرکت‌های اروپائی در این جدول، ۱۳ شرکت است که متعلق به آلمان غربی، انگلستان، فرانسه، ایتالیا، هلند و سوئد می‌باشند. رتبه هوش شرکت در سال ۸۴ و نیز مقایسه آن با سال ۸۳ به همراه اطلاعات مالی آنها در جدول مزبور نشان داده شده است. جدول (۲) صرفا شامل شرکت‌های اروپائی می‌باشد که نام برخی از آنها (که حداقل درآمدشان ۲۲۰ میلیون دلار بوده است) در جدول ۱ نیز دیده می‌شود. ترتیب قرار گرفتن نام این شرکت‌ها، میزان درآمد سالانه آنها از بازار جهانی محصولاتشان می‌باشد. لازم به یادآوری است که منظور از صنعت داده پردازی در این بررسی، تولید و عرضه کامپیوترهای بزرگ، کامپیوترهای کوچک و

ریز کامپیوترها و نیز سیستم‌های دفتری، ارتباطی، دستگاه‌های جانبی، خدمات نرم‌افزاری، نگهداری و تعمیر می‌باشد. ناگفته نماند که برخی از شرکت‌های بزرگ مانند آی تی تی و ام‌دال از جدول سال ۱۹۸۴ حذف گردیده و برخی دیگر مانند BASF راه خود را به این جدول باز نموده‌اند. در ضمن، شرکت‌های ژاپنی هنوز هیچ‌کدام نتوانسته‌اند جایی در این جدول برای خود باز کنند هرچند که احتمال راه یافتن آنها به بازار اروپا در آینده‌ای نه چندان دور، بسیار زیاد است.

در حاشیه این گزارش

۱- شرکت فیلیپس (هلند) و شرکت سی دی سی جهت ایجاد سیستم‌های ذخیره نوری، دست به همکاری زده‌اند. همچنین شرکت فیلیپس و شرکت زیمنس به منظور گسترش تکنولوژی نیمه هادی‌ها پروژه مشترکی به نام "مگا" را آغاز نموده‌اند.

۲- شرکت‌های بال (فرانسه)، آی سی ال (انگلستان) و زیمنس (آلمان) از سال گذشته مرکزی به نام مرکز تحقیقات کامپیوتری اروپا تشکیل داده‌اند تا در زمینه هوش مصنوعی تحقیق نمایند. هدف اصلی این مرکز برابری در قبال محصولات آی بی ام در زمینه هوش مصنوعی در بازارهای اروپا است. اما باید خاطر نشان ساخت که به دلیل کمبود پرسنل و عدم تصمیم‌گیری‌ها هماهنگی، این مرکز تاکنون نتایج چشمگیری به دست نیاورده است.

۳- شرکت بال، بازاریابی کامپیوترهای بزرگ (ژاپن) را به عهده گرفته است.

۴- شرکت‌های BASF و اولیوتی سال‌هاست که محصولات هیتاچی را عرضه می‌کنند.

۵- شرکت‌های زیمنس و آی سی ال، محصولات فوجیتسو را بازاریابی می‌نمایند.

به نقل از
DATAMATION
AUGUST 1985

پی‌داشته‌ها

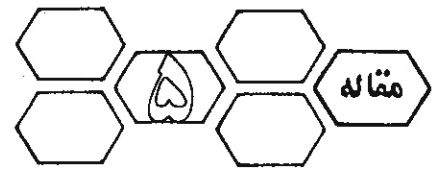
1. BULL
2. ECRC (European Computer Research Center).

FIG. 2
LEADING EUROPEAN DP COMPANIES

COMPANY	COUNTRY	WORLDWIDE DP REV. (\$ MIL.)	% CHANGE IN AAC
1 Siemens	W. Germany	2,789.5	10.6
2 Olivetti	Italy	2,012.4	28.2
3 Bull	France	1,555.6	16.8
4 ICL	U.K.	1,222.7	8.3
5 Nixdorf	W. Germany	1,147.4	20.7
6 L M Ericsson	Sweden	1,123.3	24.7
7 Philips	Netherlands	1,090.3	12.2
8 Triumph-Adler	W. Germany	631.6	15.0
9 Racal	U.K.	525.8	30.0
10 Mannesmann Kienzle	W. Germany	343.0	21.3
11 BASF	W. Germany	298.2	21.4
12 Plessey	U.K.	290.0	14.8
13 Thorn EMI	U.K.	267.3	28.2
14 Ferranti	U.K.	266.0	36.2
15 Rank Xerox	U.K.	236.0	30.1
16 Cap Gemini Sogeti	France	206.0	28.6
17 Nokia	Finland	181.0	28.8
18 Scicon	U.K.	173.3	41.4
19 Cisl	France	171.6	16.0
20 Norsk Data	Norway	166.7	53.7

1984 RANK	1983 RANK	COMPANY	HEAD- QUARTERS	1984 ERP. DP REV. (\$ MIL.)	% CHANGE (\$ MIL.)	TOTAL DP REV. (\$ MIL.)	% CHANGE (\$ MIL.)	1984 TOTAL REV. (\$ MIL.)	NET INCOME (\$ MIL.)	1984 EMPL.	FISCAL YEAR END
1	1	International Business Machines	U.S.	10,984.4	5.9	44,292.0	21.3	45,937.0	6582.0	394,930	Dec
2	2	Siemens AG	W. Germany	2,092.1	4.5	2,789.5	-1.1	16,076.8	374.0	319,000	Sept
3	4	Ing. C. Olivetti & Co. S.P.A.	Italy	1,504.6	6.2	2,012.4	10.7	2,605.6	202.6	47,613	Dec
4	5	Digital Equipment Corp.	U.S.	1,464.1	29.0	6,230.0	29.0	6,230.0	487.0	88,600	June
5	3	Groupe Bull	France	1,440.5	1.8	1,555.6	1.8	1,555.6	-55.9	26,435	Dec
6	8	LM Ericsson	Sweden	1,011.0	15.6	1,123.3	15.6	3,545.1	53.7	70,000	Dec
7	11	Hewlett-Packard Co.	U.S.	992.0	36.2	3,400.0	36.2	6,297.0	665.0	80,000	Dec
8	7	Nixdorf Computer AG	W. Germany	975.3	3.1	1,147.4	7.9	1,147.4	42.4	20,190	Dec
9	9	Burroughs Corp.	U.S.	967.5	12.5	4,500.0	12.5	4,875.6	244.9	65,000	Dec
10	6	ICL	U.K.	917.0	-4.6	1,222.7	-4.6	1,222.7	18.7	15,167	Dec
11	10	N.V. Philips Gloeilampenfabrieken	Netherlands	839.5	-4	1,090.3	-0.4	16,766.0	346.0	344,000	Dec
12	12	Sperry Corp.	U.S.	799.0	13.0	3,473.9	13.0	5,370.0	267.6	77,000	Mar
13	13	NCR Corp.	U.S.	707.7	1.0	3,670.0	10.1	4,074.3	342.6	62,000	Dec
14	15	Wang Laboratories Inc.	U.S.	435.7	37.3	2,420.7	35.0	2,421.1	230.7	30,000	June
15		Commodore International Ltd.	U.S.	406.6	92.4	1,129.5	21.8	1,189.5	100.3	NA	June
16	14	Control Data Corp.	U.S.	375.0	7.0	3,755.5	7.0	50,269.9	31.6	54,100	Dec
17	16	Mannesmann Kienzle GmbH	W. Germany	343.0	8.5	343.0	8.5	428.1	104.3	8,000	Dec
18		Apple Computer Corp.	U.S.	315.1	75.0	1,897.9	74.9	1,897.9	104.3	5,382	June
19	18	Honeywell Inc.	U.S.	310.2	9.5	1,825.0	9.5	6,073.6	239.0	94,300	Dec
20	17	Plessey Co. plc	U.K.	290.0	1.0	290.0	1.0	1,953.1	127.9	38,838	Dec
21	19	Thorn EMI plc	U.K.	242.7	12.7	267.3	7.3	3,761.2	120.8	91,544	Mar
22	21	Rank Xerox	U.K.	236.0	14.6	236.0	14.6	2,353.3	112.0	31,000	Oct
23		BASF	W. Germany	228.1	8.6	298.2	8.6	12,392.0	NA	31,000	Dec
24	22	Ferranti plc	U.K.	226.6	19.8	266.0	19.8	666.7	38.4	11,087	Dec
25	25	Data General Corp.	U.S.	220.1	34.4	1,229.7	40.9	1,229.7	78.4	17,695	Sept

NA: NOT AVAILABLE



مفاهیم اولیه سیستمهای عامل

(قسمت دوم)

ترجمه*: عباس مصطفوی سبزواری
منابع و خدمات کامپیوتری
(ایزیران)

پاداش سردبیر

در شماره گذشته، اولین بخش از مقاله تفصیلی "مفاهیم اولیه سیستمهای عامل" که ترجمه بخش سیستمهای عامل دانشنامه AÜERBACH می باشد، از نظر خوانندگان گرامی گذشت. این مقاله جلب می نماید.

* * *

تخصیص منابع^۱

تخصیص منابع سیستم از قبیل حافظه، دستگاههای ورودی و خروجی، زمان پردازنده، و پروندههای اطلاعاتی، توسط روالها صورت میگیرد که با توجه به نیازهای برنامه، منابع مورد نیاز را به نحوی که بهترین استفاده از آنها به عمل آید، تخصیص میدهند. معمولاً نیازهای برنامه به منابع، یا توسط استفاده کننده، یا از طریق خود برنامه، و یا با استفاده از زبان کنترل کار (JCL) اعلام میگردد.

در پردازش ردیفی و پیاپی، تخصیص منابع به سادگی انجام میگیرد زیرا هر برنامه در زمان اجرا، تمام منابع موجود را جهت استفاده در اختیار دارد. اما در محیطهای چندبرنامه ای و اشتراک وقت^۲، این امر نیازمند روالهای پیچیده تری است تا از ایجاد تداخل در تخصیص منابع (مخصوصاً موقعی که منابع عمومی مورد استفاده قرار میگیرند)، جلوگیری شود. بنابراین، بسیاری از الگوریتمهای زمان بندی بر این اساس بنا شده اند که تمام منابع مورد نیاز یک کار، قبل از شروع اجرای آن، بطور ثابت به آن تخصیص داده شده باشند. در استفاده از این روش (تخصیص ایستا^۳)، مزایا و معایبی نهفته است. مزیت موجود بودن تمام منابع این است که کار می تواند بدون تأخیر و انتظار برای آزاد شدن منابع مورد نیاز، تا انتها ادامه یابد. اما اشکال اصلی این روش در آن است که برنامه های بعدی ناپستی تا پایان یافتن برنامه های قبلی (حتی اگر نیازهای برنامه ها به منابع برطرف شده باشد)، انتظار بکشند. بعضی از سیستمها، اجازه شروع کار را تنها با حداقل منابع موجود می دهند. اشکال این کار در این است که اگر سیستم قادر به پیش بینی و آینه نگری درباره منابع مورد نیاز بعدی نباشد، عملاً کارآیی خود را از دست

خواهند داد. در این حالت، سیستم با پیداکارهایی را که نمی تواند منابع مورد نیازشان را تأمین کند، تا زمان آزاد شدن این منابع، در یک صف انتظار قرار دهد.

به تعویق انداختن کارها، تنها اشکال این روش نیست. بلکه در صورتی که به برنامه ها این اجازه داده شود که منابع مورد نیاز خود را تا انتهای اجرا نگه دارند، این اشکال به وجود می آید که هرگاه برنامه ای به دلایلی به حالت تعلیق درآید، تمام برنامه های بعدی به طریق اولی از استفاده از منابعی که در اختیار آن برنامه قرار دارند، محروم خواهند شد. و مسلم است که در حالتیکه تعداد این نوع برنامه ها زیاد باشد، توان عملیاتی سیستم به سرعت تحلیل خواهد رفت.

بسیاری از اشکالات فوق با انتخاب روشی به نام "تخصیص پویای منابع"^۵ از بین خواهد رفت. در این روش نه تنها کلیه منابع مورد نیاز مورد توجه قرار میگیرند، بلکه به برنامه ها در حال اجرا اجازه داده می شود تا در همان زمانی که به منابعی نیاز دارند، آن را از سیستم تقاضا کنند. سیستم این گونه درخواستها را از مخزن عمومی منابع برطرف می سازد و هرگاه منابع به یک منبع از بین برود مجدداً آن را به مخزن برمیگرداند.

طریقه "تخصیص پویای منابع" مخصوصاً برای سیستمهای "چندبرنامه ای" پیشرفته و در مواقعی که تعداد برنامه های در حال اجرا روبه افزایش باشد، حداکثر کارآیی خود را ارائه خواهد داد. در این روش، چون تخصیص منابع فقط به لحاظ استفاده محدود می گردد (در مقابل روش "ایستا" که منابع برای تمام طول اجرا در اختیار برنامه قرار میگیرند)، میزان کسل منابع مورد نیاز برای یک سیستم به تنهایی چشمگیری کاهش میابد. البته در اختیار داشتن این روش، هم به هزینه سربار^۶ سیستم عامل می افزاید و هم امکان پدید آمدن حالت بن بست^۷ را افزایش میدهد.

همانطور که قبلاً بیان گردید، منابع اصلی یک سیستم عبارتند از: حافظه اصلی، دستگاههای ورودی و خروجی، زمان پردازنده، و پروندههای اطلاعاتی. در زیر به شرح مختصر چگونگی تخصیص هر کدام میپردازیم.

الف - حافظه اصلی: در محیطهای چندبرنامه ای پنج روش اصلی برای تخصیص حافظه اصلی وجود دارد:

۱ - تخصیص ایستا

در این روش مقدار ثابتی از حافظه برای پردازش اصلی^۸ (با اولویت بیشتر) و پردازش جنبی^۹ (با اولویت کمتر) در اختیار برنامه ها قرار میگیرد. محدودهای که برای پردازش اصلی (با اولویت بیشتر) تخصیص میابد معمولاً برای پردازشهای بلند رنک^{۱۱} و با اشتراک و قسمت و محدود جنبی، برای پردازش دسته ای^{۱۲} مورد استفاده قرار میگیرد. این گونه ها زمان دهی

حافظه، از این واقعیت سرچشمه میگیرد که اولویت برنامه توسط تخصیص حافظه معین می گردد. لذا محدودهای اصلی همواره نسبت به محدودهای جنبی دارای اولویت اجرایی میباشند.

۲ - تقسیم بندی ثابت^{۱۳}

در این روش، حافظه از تعدادی بخشهای مستقل با اندازه های مختلف تشکیل میگردد. هر بخش می تواند در جریان ورودی جداگانه ای داشته باشد یا این که هر برنامه می تواند کوچکترین بخش قابل انتخاب را که برای اجرای کافی باشد، اختیار کند. در حالت اخیر، سیستم میتواند دارای پارامترهایی باشد که از تخصیص بعضی از بخشها به برنامه های خاصی جلوگیری نماید.

۳ - مخزن حافظه

در این روش، کل حافظه موجود در سیستم، به صورت یک مخزن اشتراکی مورد استفاده قرار میگیرد. بدین صورت که هر برنامه به حسب نیاز خود، مقدار رپیوسته ای از حافظه را برای انجام کارهای خود انتخاب کرده و پس از رفع نیاز آن را به مخزن برمیگرداند. اگر عملیات در محیطهای اشتراک وقتی صورت گیرد، معمولاً یک بلوک حافظه برای رفع نیاز تمام کاربردهای اشتراک وقتی از مخزن خارج میگردد.

۴ - صفحه بندی^{۱۴}

در این طریقه، حافظه اصلی به بخشهای با اندازه نسبتاً کوچک تقسیم میگردد. این بخشها که به آنها صفحه گفته میشود، در یک مخزن اشتراکی قرار دارند و داده ها و دستورالعملهای برنامه منبزه سهم خود به صفحاتی تقسیم می شوند.

موقعی که حافظه اصلی به تعدادی صفحه تقسیم شده باشد، با رکورد یک برنامه می تواند با از طریق معمول و با رکورد تمام صفحات مورد نیاز حافظه اصلی صورت گیرد، و با از طریق فراروندی که به آن "صفحه بندی" گفته میشود. در روش صفحه بندی، به یک حافظه جانبی با دستیابی سریع (معمولاً دیسک یا استوانه مغناطیسی^{۱۵}) جهت پشتیبانی حافظه اصلی نیاز خواهد بود. حافظه جانبی حافظه اصلی با پدیده دوبه صفحاتی با اندازه های یکسان تقسیم شده باشند.

برنامه ها که مل بر روی حافظه جانبی فقط صفحهای از آن که دستورالعملهای آن باید در آن لحظه اجرا گردند در حافظه اصلی قرار میگیرد. هرگاه در درون اجرای برنامه به صفحه دیگری نیاز بود، صفحهای در حافظه اصلی برای این منظور آماده می گردد - احتمالاً با انتقال صفحه قبلی به حافظه جانبی - و صفحه مورد نیاز به داخل حافظه اصلی فرا خوانده میشود. در این سیستم، تصویری از صفحات تهیه می گردد که نشانگر نشانی فیزیکی صفحات (در حافظه اصلی یا حافظه جانبی) و نشانی واقعی ورق بزنید

حافظه (محل خانه‌های حافظه اصلی یا شماره شیار ۱۶ در حافظه جانبی) می‌باشد. از این تصویر برای نشان‌دهی و دستیابی به دستور-العملها و داده‌های داخل صفحات استفاده می‌شود. در بسیاری از سیستمها این مسأله از طریق مدارهای سخت‌افزاری و در برخی سیستمها نیز به وسیله نرم‌افزار صورت می‌گیرد.

با استفاده از تکنیک صفحه‌بندی می‌توان هر نوع برنامه‌ای را حتی با طول نسبتاً نامحدود اجرا نمود زیرا که در این روش عملاً هیچگاه به بازرگدن تمام برنامه در حافظه اصلی نیازی نخواهد بود و محدودیت اندازه حافظه اصلی، مسأله‌ای را به وجود نمی‌آورد. از طرف دیگر، به علت انتقال مرتب صفحات بین حافظه اصلی و جانبی، هزینه سربار سیستم افزایش خواهد یافت. در موقعی که سیستم در حال انتقال صفحات بین دو حافظه است، عملاً یا کار دیگری انجام نمی‌دهد و یا زده‌موش آن بسیار کم خواهد بود. به این حالت، "هرز رفتن" گفته می‌شود.

روشهای چندی برای به حداقل رساندن انتقال صفحات بین حافظه اصلی و جانبی و نتیجتاً کاستن از هرز رفتن و احداث زنده‌مرکزی ارائه گردیده‌اند. یکی از روشهای معمول این است که صفحه‌ای را که احتمالاً به جعبه‌ت در بقیه مدت اجرا از باقی صفحات کمتر است، مشخص ساخته و در اولین نیاز به جایگزینی، جای آن را به صفحه جدید بدهیم. روش دیگر این است که تنها صفحاتی را به حافظه جانبی انتقال دهیم که از ابتدای قرار گرفتن در حافظه اصلی، اصلاحاتی روی آن صورت نگرفته باشد. مزیت این کار در این است که دیگر نیازی به نوشتن محتویات این چنین صفحاتی در حافظه جانبی وجود ندارد زیرا عین محتویات آنها در آنجا وجود دارد.

بعد از تخصیص حافظه مورد نیاز، معمولاً برنامه‌ها به حافظه بیشتری برای میانگیریها توسعه برنامه‌ها و غیره نیاز دارند. در اغلب سیستمها این تخصیص حافظه به صورت پویا انجام می‌گیرد بدین نحو که حافظه مورد نیاز از یک مخزن اشتراکی که مستقل از تخصیصهای عادی حافظه است گرفته شده و پس از استفاده، یا صریحاً توسط خود برنامه‌ها یا به طور ضمنی پس از اتمام برنامه به مخزن برگردانده می‌شوند. استفاده از روش صفحه‌بندی دارای مزیت دیگری نیز می‌باشد که آن، از بین بردن تقریبی پدیده تکه‌تکه شدن حافظه است. این پدیده هنگامی رخ می‌دهد که به دلیل توزیع غلط برنامه در طول حافظه و یا عدم وجود تکنیکهای فشرده‌سازی حافظه ۲۰ در سیستم، قطعات آزاد حافظه با اندازه‌های متفاوت در طول آن پراکنده گردند و به دلیل کوچکی اندازه نتوانند مورد استفاده برنامه جدیدی واقع گردند. البته وقوع این پدیده در محیطهای اشتراکی که حافظه به بخشهای برنامه‌ها و داده‌ها و متفاوت تقسیم شده باشد.

به عنوان مثال، فرض کنید در یک سیستم، کل حافظه شامل ۱۰۰ کیلوبایت برای اجرای برنامه‌ها، مختلف با طولهای ۲۰، ۶۰ و ۲۰ کیلوبایت به ترتیب از بالا به پایین اشغال شده باشد. حال اگر فرض کنیم برنامه‌های اول و سوم با سرعت بیشتری نسبت به برنامه دوم تکمیل شوند، در صورتی که سیستم عامل قادر به انجام عمل "فشرده‌سازی حافظه" نباشد، کل حافظه رها شده که ۴۰ کیلوبایت است به دو قسمت جداگانه هر کدام با طول ۲۰ کیلوبایت تقسیم شده و این دو حافظه آزاد ۴۰ کیلوبایتی، در طرفین حافظه برجای خواهند ماند (در صورت فشرده‌سازی، مثلاً برنامه ۶۰ کیلوبایتی به یک انتهای حافظه منتقل می‌گردد و ۴۰ کیلوبایت حافظه آزاد به پیوسته در یکجا باقی می‌ماند و احتمالاً از هر دو رفتن منابع جلوگیری می‌شود). حال اگر برنامه‌های با طول بیشتر از ۲۰ کیلوبایت (مثلاً ۳۰ کیلوبایت) نیاز به حافظه برای اجرا داشته باشند، در عین حال که به اندازه مورد نیازش حافظه وجود دارد، لیکن به دلیل پراکندگی حافظه موجود، نمی‌توانند از آن استفاده کنند و این امر به دلیل وقوع پدیده "تکه‌تکه شدن حافظه" است.

این نقیصه در سیستمهایی که قدرت فشرده‌سازی حافظه را دارند اگرچه تا حدودی رفع می‌شود ولی باز هم نمی‌توانند به دلیل ضعف زمان بندی کارها از جانب استفاده کننده، به میزان کمی بروزشانند. بدیهی است که حتی با وجود هوشمندترین برنامه‌نویسها نیز نمی‌توان در محیطهایی که در آنها حافظه به بخشهای با طولهای متفاوت تقسیم شده باشد، از تکه‌تکه شدن حافظه جلوگیری کرد. البته در صورتی که سیستمی قادر باشد مقدار حافظه را بر حسب نیاز توسعه دهد و توانایی فشرده‌سازی نیز داشته باشد، این مسأله برطرف خواهد شد و لیکن سفا نه این قدرت در اختیار تمام دبسیار محدودی از سیستمها است.

۵ - حافظه مجازی

در این روش با استفاده از دیسکها و استوانه‌های مغناطیسی سریع، ظرفیت حافظه اصلی سیستم را افزایش می‌دهند و آن را در نظر استفاده کننده، بیش از آنچه واقعاً هست، نمودار می‌سازند. البته حافظه افزایش یافته، کندتر از حافظه اصلی خواهد بود ولی این، روشی ارزان برای توسعه حافظه اصلی می‌باشد. در کامپیوترهای مجهز به حافظه مجازی، سیستم عامل اجباراً با بستن مسئولیتهای بسیار زیادتری را نسبت به سیستمهای ساده‌مهندار شده مهم‌ترین آنها عبارت است از مدیریت حافظه اصلی که به موجب آن باید لیست نظایق یک به یک بین نشانهای حافظه مجازی در حافظه جانبی و محل‌های واقعی حافظه اصلی تهیه گردد.

حافظه مجازی می‌تواند به بندها ۲۲ یا صفحاتی تقسیم شده باشد. حافظه صفحه‌بندی شده، دارای این امتیاز است که خیال برنامه‌ها

نویس را از بیابان محدودیت حافظه در حافظه‌ها اندازه‌برنامه‌هاش، کاملاً راحت می‌کند. اندازه بخشهای حافظه مجازی را می‌توان به میزان مورد نیاز در نظر گرفت. به این جهت، برنامه‌نویسهای کاربردی می‌توانند توجه خود را بر روی کاربرد مورد نظرشان متمرکز سازند. برنامه‌نویس نی نیز که به اصلاح و نگارهای برنامه‌ها اشتغال دارند، نباید از این با بستانگران شوند که ایجا د تغییرات در برنامه‌ها و از بدین دلول آنها ممکن است باعث شود تا برنامه اصلاح شده در یک بخش حافظه مجازی جا نگیرد. در این مورد کافی است برنامه‌نویس بدون هیچگونه شکالی فقط طول بخش حافظه را زیاد کند. به علاوه، برنامه‌نویسهای سیستم نیز دیگر نیازی به یکا رگیری تکنیکهای واحدی کردن آنها و ایجا د ارتباط و اتصال بین واحدها نخواهند داشت و می‌توانند خود را به مسأله مهمتری از قبیل مدیریت و لوییت کارها معطوف دارند.

تنها اشکال حافظه مجازی صفحه‌بندی شده این است که به بالا رفتن تعداد انتقالها بین حافظه اصلی و جانبی ممکن است سیستم به حدی درگیر کارهای انتقالها گردد که از کارایی خود بازماند.

حافظه مجازی بندبند شده زندهای متغیری از حافظه اصلی و مجازی که بر اساس نقاط تقسیم منطقی برنامه‌ها بنا شده‌اند، استفاده می‌کند. اشکال اساسی این روش در این است که در صورت اختلاف طول در بندها (که معمولاً حاصل عدم دقت برنامه‌نویس در طراحی برنامه و نتیجتاً عدم تقسیم صحیح برنامه‌بندهای هم‌زمانی‌ها شد)، کارآیی سیستم به سرعت پائین می‌آید. دلیل این امر این است که سیستم مجبور است برای اجرای برنامه‌ها و هموار شده اندازه طول بزرگترین بند روی حافظه اصلی جا ذخیره داشته باشد. به هر حال، در صورتی که برنامه‌نویس نتواند برنامه‌ها خود را در بندهای کم‌بیش مسأله طراحی کند، استفاده از این روش مقرون به صرفه نخواهد بود. و بندهای کوچک، باعث اتلاف حافظه می‌گردند. البته بعضی از کامپیوترها با روش اخیر کاملاً هماهنگ هستند و استفاده مؤثری از این روش می‌نمایند. بهر حال، به دلیل این که اغلب طول بندها در مقایسه با صفحات بزرگتر است، هرز رفتن حافظه کمتر است اتفاق می‌افتد.

از نظر تئوری، هر سیستمی می‌تواند به طور نرم افزاری، حافظه مجازی را پیاپی ده‌کند. تنها کاری که باید انجام گردد، پیگیری ردیابی نشانها و ایجا د جدول مرجع و برنامه‌های مهم تطبیق دهنده است. هنگامی که جنبه‌های مهم عملیاتی چنین سیستمهایی به صورت سخت‌افزاری پیاپی ده‌شده باشد، تولید کنندگان ادعا می‌کنند که یک سیستم حافظه مجازی ایجاد نموده‌اند. استفاده کنندگان نباید از این گونه مفاهیم سیستمهای حافظه مجازی گمراه

بطورداشم روی سیستم قرار داده شوند (مورد اخیر وقتی است که روال به دفعات مورد استفاده واقع شود).

نوع تخصیص روالها به برنا مه نسبت به نوع و ماهیت روال فرق می کند. روالهای "یکبار مصرف" را با بستی هر موقع مورد نیاز باشند، از ابتدا "بار" کرد. روالهای "چندبار مصرف" را لازم نیست به تعداد دفعات مورد نیاز "بار" نمود ولی به دلیل احتمال تغییر مقدار ثابت یا دستورالعملها، در هر لحظه آنها در یک برنا مه می توان نمود مورد استفاده قرار گیرند. روالهای "چنددخولی" ۲۶ را می توان در هر لحظه توسط برنا مه های مختلف مورد استفاده قرار داد. هم انطور که مشاهده می شود، تخصیص روالهای "چنددخولی" نسبتا ساده است ولی در مورد روالهای چندبار مصرف با بستی امکانناقصی برای محدود ساختن تخصیص آنها به یک برنا مه در هر زمان، وجود داشته باشد.

ساختن برنا مه

اغلب سیستمهای عامل روشهای مختلفی را جهت "بار کردن" برنا مه ها مورد استفاده قرار می دهند که معمولترین این روشها عبارتند از: بار کردن ساده، جایگذاری، پویا، و صفحه بندی شده. این روشها با روشهای مربوط به تخصیص حافظه که قبلا" مورد بحث قرار گرفتند مشابهت زیادی دارند. روش ساده، شامل یک واحد بار کردن ۲۷ منفرد است که مقدار ثابتی از حافظه را اشغال می کند (واحد بار کردن که بعدا" مورد بحث قرار خواهد گرفت، شامل مجموعه ای از مجموعه دستورالعملها است).

در روش جایگذاری، استفاده مجدد از بلوکهای مشابهی از حافظه اصلی در خلال مراحل مختلف اجرای برنام، مجاز است. بنا بر این برنا مه به چندین تقسیم می گردد و هر یک در صورت نیاز، به حافظه آورده می شود.

روشهای پویا و صفحه بندی شده، اصولا" روشهای پیچیده ای هستند و فقط در سیستمهای بزرگ "چند برنا مه ای" و "با اشتراک وقت" معمول هستند. در روش بار کردن پویا، یک واحد جای بستی پذیر ۲۸ (یعنی واحدی که بتواند در محل مفروضی از حافظه بار گردد)، در صورت نیاز برنا مه در حال اجرا به آن، به حافظه اصلی بار می گردد. اختلاف بار کردن پویا با روش جایگذاری در این است که اولاً لازم نیست محل لازم برای واحد در حافظه اصلی از قبل مشخص گردد و ثانیاً "الزاما" این واحد، محل واحد دیگری را در حافظه اشغال نخواهد کرد.

یکی از مسائل عمده ای که در رابطه با روش بار کردن پویا وجود دارد، به وجود آمدن پدیده تکه تکه شدن حافظه است. این پدیده هنگامی به وجود می آید که تمام برنا مه های که به صورت پویا بار شده اند، بطور همزمان خاموش می شوند.

و ر ق ب زنید

در یافت سرویس از سوی برنا مه ها داده می شود در این لحظه، برنا مه در "صف" دیگری که معمولاً صف انتظار نامیده می شود، قرار گرفته و برنا مه بعدی از صف ورودی، پردازنده را در اختیار می گیرد. حال اگر عمل سرویس دهی برای برنا مه اولی پایان یابد، اجرای برنا مه دوم قطع می گردد و آن برنا مه در لیست انتظار قرار می گیرد و پردازنده مجدداً در اختیار نخستین برنا مه گذاشته می شود.

از نقاط ضعفی که در این روش وجود دارد این است که اصولاً تا برنا مه ای با اولویت اجرای بالا اجرا نگردد و یا کلاً در لیست انتظار قرار نگیرد، برنا مه ای با اولویت کم اجرا نخواهند شد و بهر حال در تعدادی از کاربردها این تاءخیر قابل پذیرش نیست. مسأله دیگر این است که مقدار زمانی که پردازنده در اختیار همه کارها گذاشته می شود، نامعین است.

برای حل این مشکلات، روشهای چنبدی وجود دارد که از همه مهمتر، انتخاب بیشتر از یک صف اجرای، تخصیص رده معینی به هر صف، و انتخاب تکه های زمانی ۲۵ متفاوت برای رده های مختلف می باشد. این روش معمولاً با تعریف بخشهای ثابت در حافظه انجام می شود و برنا مه های موجود در رده های مختلف به محلهای متفاوتی از حافظه مراجعه می کنند. عمل اجرا فقط در طول تکه زمانی مربوطه انجام شده و بنا بر این، برنا مه ای با اولویت بیشتر، زمان بیشتری از پردازنده را در اختیار خواهد داشت و عملاً فلسفه وجود رده های مختلف اجرای سی صورت واقعیت می گیرد. از طرفی به دلیل آن که برنا مه ها به بخشهای مختلف حافظه وابسته می شوند، عملاً شانس معقولی برای همه آنها جهت در اختیار گرفتن پردازنده وجود دارد.

ت - پرونده های اطلاعاتی

منظور از پرونده های اطلاعاتی می تواند پرونده ای کتابخانه برنا مه ها و زیر روالهای مختلف باشد که تخصیص هر کدام از آنها با روش جداگانه ای انجام می شود.

پرونده اطلاعاتی را معمولاً می توان در خلال پردازش زمان بندی، شبیه دستگاهی ورودی/خروجی به صورت ایستاد در اختیار برنامها قرار داد. در مورد سیستمهای چند برنا مه ای نیز منابع بر حسب نوع نیاز برنا مه ها (فقط خواندن و یا فقط نوشتن) می توانند بطور پویا به برنامها های مختلف وابسته گردند.

برنا مه ای زیر روالها می توانند بر حسب نیاز بطور انحصاری به برنا مه خاصی وابسته شده و یا مشترکاً توسط برنا مه های مختلف مورد استفاده واقع شوند. روالهای اشتراکی بجای آن که نظیر روالهای مستقل تمام برنامها خاص وابسته باشند می توانند مشترکاً فقط در موقع نیاز بار گرفته شوند. این روالها می توانند دروی حافظه ای جانبی مشخصی قرار گیرند، در هر محل خاصی از حافظه واقع شوند، و یا با اخر به صورت جزئی از سیستم مقیم حافظه

گردند و یا بدیهه خاطرداشته باشند که استفاده از حافظه جانبی یا دستیابی مستقیم برای پیاده کردن حافظه مجازی، ۳ تا ۵ برابر کندتر از حافظه اصلی کامپیوتر است. استفاده کنندگان همچنین باید در نظر داشته باشند که مدت اجرای برنا مه ها در سیستمهای حافظه مجازی نسبت به سیستمهای احتمالاً بیشتر است. مزیت استفاده از سیستمهای حافظه مجازی در واقع در محیطهای چند برنا مه ای است که می توان عملیات سیستم را افزایش می دهد.

ب - دستگاههای ورودی/خروجی

تکنیکهای که به منظور تخصیص دستگاههای ورودی/خروجی وجود دارند، بسته به محیطهای پردازشی مختلف، متفاوت هستند. به عنوان مثال، در سیستمهای پردازش اصلی (با اولویت بیشتر) / جنبی (با اولویت کمتر)، دستگاههای ورودی/خروجی بطور ثابت به یکی از قسمتهای اصلی یا جنبی اختصاص می یابد. بطوری که انجام هرگونه تغییر بعدی در آن، مستلزم دخالت اپراتور خواهد بود. از طرف دیگر در سیستمهای بلادرنگ، کنترل تخصیص دستگاههای ورودی/خروجی وجود ندارد و برای جلوگیری از تصادم در یکا رگیری دستگاهها، باید از قرار دادهای برنا مه نویسی استفاده شود.

در سیستمهای که از روش صفحه بندی و بخش کردن حافظه استفاده می کنند، تخصیص دستگاههای ورودی/خروجی می تواند به دو صورت ایستاد یا پویا باشد. معمولاً در سیستمهای که دستگاههای ورودی/خروجی ردیفی مانند کارت خوان استفاده می نمایند، از تخصیص ایستاد در سیستمهای استفاده کننده از منابع دستگاهی دستیابی تصادفی، از تخصیص پویا استفاده می شود.

پ - زمان پردازنده

تخصیص زمان پردازنده در محیطهای چند برنا مه ای و با اشتراک وقت از طریق روال "توزیع کننده" امکانات ۲۴ کیکی از عناصر اصلی سیستمهای عامل است انجام می پذیرد. "توزیع کننده" امکانات، مسئول برنامها می باشد که در انتظار پردازش بر سر می برد و تخصیص پردازنده به آنها است.

معمولاً در یک پردازش دسته ای، کلیه برنامها جهت اجرای در یک صف اجرای به نام "صف توزیع کننده" امکانات قرار می گیرند و برنا مه ای با اولویت اجرای بالاتر، محل جلوتری را در صف اشغال می نمایند. در صورتی که لازم باشد برای شروع اجرا، تمام منابع مورد نیاز در اختیار برنا مه قرار می گیرند. این تخصیص منابع با بستی قبل از قرار گرفتن برنا مه در صف اجرای مذکور انجام می شود.

معمولاً توزیع کننده امکانات، اولین برنا مه موجود در "صف" را برای اجرا انتخاب می کند و پردازنده را در اختیار آن قرار می دهد. عمل اجرا تا هنگام دریافت تقاضای مبنی بر

تولید و احداث رکوردی

اغلب کامپایلرها و اسمبلرها برنامه‌های را که شده را مستقیماً به حالت قابل اجرا تبدیل نمی‌نمایند بلکه برنامه‌ها ابتدا به صورت کد جا بجا می‌پذیرد و در ادامه و سپس با اعمال ملاحظات دیگری روی آنها (معمولاً) از بین بردن عدم کارایی‌های (مخصوصاً) آماده اجرا می‌گردند. به عنوان مثال، ممکن است برنامه، حافظه مورد نیاز را برای خود را در اختیار نداشته باشد. به فراروند تبدیل این گونه برنامه‌ها به شکل قابل بار شدن و اجرا، "تولید و احداث رکوردی" گفته می‌شود.

بسیاری از سیستم‌ها می‌توانند واحدهای کد جا بجا می‌پذیری که توسط کامپایلرها یا مختلف تولید شده اند را با زیررواها و سایر واحدهای کد جا بجا می‌پذیر ترکیب نموده، یک واحد بار کردنی منفرد به وجود آورند. در این حالت، این واحدها معمولاً به مکان‌های متوالی و مجاور حافظه تخصیص داده می‌شوند و کلیه نشانی‌های داخلی برنامه‌ها بر اساس محل قرار گرفتن روی حافظه، اصلاح می‌گردد. درحالی‌که لازم باشد قسمتهای مستقل یک یا چند برنامه به یکدیگر پیوند یافته و توالی اجرا گردند، تغییرات لازم نشانی با توجه به محل قرار گرفتن هر قسمت از برنامه در حافظه اصلی انجام می‌گیرد. مکان نیزم خیر در سیستم‌های مختلف به نام‌های مختلفی نامیده می‌شود که مشهورترین آنها ویرایش پیوندی^{۲۹} نام دارد.

تعداد زیادی از سیستم‌ها قابلیت اضافه کردن ضمنی روال‌های جا بجا می‌پذیرا دارند بدین معنی که اگر در پایان مرحله موسوم به ویرایش پیوندی ملاحظه شود که تعدادی از روال‌های تقاضا شده توسط برنامه را نداشته‌اند، سیستم بطور ضمنی فرض را بر این قرار خواهد داد که بایستی این روال‌ها را در کتابخانه سیستم جستجو نماید. این روش مخصوصاً در برنامه‌های مولدکد (نظیر کامپایلرها، سیستم‌های مدیریت داده‌ها، و تولیدکننده‌های گزارش) که در آنها یک لیست مرصع از روال‌های پشتیبانی کننده در خلال مرحله تولید کد لازم نیست، بسیار مؤثر است.

کنترل برنامه

درجه کنترل سیستم عامل بر روی اجرای برنامه‌ها در سیستم‌های مختلف با یکدیگر متفاوت است. در اغلب آنها پیش‌بینی اموری از قبیل کنترل توزیع امکانات، کنترل پردازش وقفه^{۳۰}، همزمان ساختن رویدادها^{۳۱} و در نظر گرفتن محدودیت برنامه‌ها انجام می‌گیرد. کنترل توزیع امکانات هم‌انطور که قبلاً هم مورد بررسی قرار گرفت به منظور تخصیص وقت پردازنده به برنامه‌های مختلف انجام شده و پردازش وقفه، برای پاسخگویی به رده‌های متفاوت وقفه‌ها انجام می‌گیرد. معمولاً وقفه‌ها در موارد زیر به وجود می‌آیند:

• درخواست ورودی/خروجی

• بروز خطا در برنامه

• بروز خطا در ماشین

• وقفه‌های صادر شده توسط اپراتور

از نظر کلی، وقفه‌ها را می‌توان به دو نوع سطح بالا و سطح پایین تقسیم نمود. اگر وقفه از نوع اول باشد، بلافاصله جریان عملیات اجرا می‌قطع و پردازنده را به اجرای خود اختصاص می‌دهد. وقفه‌های سطح پایین معمولاً معوق گذاشته می‌شوند. در صورتی که تعدادی وقفه به طور همزمان پیش‌آیند، به صورت عادی در یک صف قرار داده می‌شوند و به ترتیب اولویت، مورد پردازش قرار می‌گیرند.

به تعبیر دیگر، وقفه‌ها را می‌توان به دو نوع با نقاب^{۳۲} و بی‌نقاب^{۳۳} تقسیم نمود.

در حالت اول، در لحظه دریافت وقفه، به پردازنده این فرصت داده می‌شود که آخرین دستورالعمل در حال اجرا را تکمیل نماید و سپس برنامه و محتویات آن را به همراه محتویات ثبت‌ها در محلی ذخیره کرده و به اجرای دستورالعمل‌های خواسته شده توسط وقفه بپردازد. حال اگر در حین اجرای روال‌های مربوط به این وقفه مجدداً وقفه دیگری با اولویت بالاتر به سیستم وارد شود، عین فعل و انفعالات قبل انجام شده، وقفه در حال اجرا با اولویت کمتری در نوبت قرار گرفته، و اجرای روال‌های مربوط به وقفه جدید شروع می‌گردد. این امر تا پایان اجرای تمام وقفه‌ها ادامه یافته و سپس کنترل از طریق سیستم عامل به روال توزیع کننده امکانات منتقل می‌گردد.

تکنیک همزمان ساختن رویدادها معمولاً این مفهوم را دارنده اجرای یک برنامه در لحظه بروز رویداد خاصی به تاءخیر یا فست‌دو یا اجرای یک برنامه در لحظه بروز رویداد خاصی آغاز گردد. متداولترین انواع رویدادها که می‌توانند باعث به تاءخیر افتادن یا راه اندازی اجرای یک برنامه گردند، عبارتند از:

• تکمیل عملیات ورودی/خروجی

• خاتمه تکه‌های زمانی انتخاب شده

• و تکمیل برنامه‌های فرعی وزیررواها

همزمان ساختن رویدادها می‌تواند بین قسمتهای مختلف یک برنامه برقرار گردد. یک برنامه می‌تواند تعداد زیادی برنامه فرعی دیگر را در برداشته باشد که هر کدام از آنها به تنهایی و یا بطور همزمان با دیگر برنامه‌ها و یا خود برنامه اصلی اجرا گردند و برنامه اصلی می‌تواند در هر لحظه با صدور درخواست "انتظار"، اجرای بقیه برنامه‌ها را تا تکمیل یک یا چند برنامه فرعی به حالت تعلیق درآورد.

در نظر گرفتن محدودیت برنامه‌ها عبارت است از کنترل لحظه‌ای منابع محدود درخواست شده به وسیله هر برنامه و تطبیق آن با مقدار درخواست شده توسط برنامه. این منابع معمولاً عبارتند از: زمان پردازنده، تعداد درکورد های تولید شده خروجی، مقدار کل حافظه اصلی یا

جانبی تخصیص یافته. بعضی از سیستم‌ها به محض رسیدن به حد نامی استفاذه از این منابع، برنامه‌ها را بطور خودکار ختم می‌نمایند در حالی که در بعضی از سیستم‌ها چگونگی ادا برنامه‌ها از استفاذه کننده پرسیده می‌شود.

(پایان قسمت دوم)

پیادداشتها

- 1: Resource Allocation
- 2: Multiprogramming
- 3: Time-sharing
- 4: Static Assignment
- 5: Dynamic Resource Allocation
- 6: Pool
- 7: Overhead
- 8: Deadlock
- 9: Foreground Processing
- 10: Background Processing
- 11: Real-time
- 12: Batch Processing
- 13: Fixed Partition
- 14: Paging
- 15: Drum
- 16: Track
- 17: Thrashing
- 18: Buffers
- 19: Memory Fragmentation
- 20: Memory Compression
- 21: Virtual Memory
- 22: Segments
- 23: Modularization
- 24: Dispatcher
- 25: Time Slices
- 26: Reentrant
- 27: Load Module
- 28: Relocatable Module
- 29: Linkage Editing
- 30: Interrupt Processing Control
- 31: Event Synchronization
- 32: Masked
- 33: Unmasked

مواظت و بهای درج آگهی در گزارش کامپیوتر

اندازه‌های مجاز (قبل از کوچک شدن):

پهنای ۸، ۱۷، با ۲۶ سانتیمتر
بلندی - هر مقدار تا حداکثر ۳۵ سانتیمتر

بهای درج برای یک نوبت

هر سانتیمتر بلندی در عرض یک ستون ۱۰۰ ریال
ربع ستون (۸×۸) سانتیمتر " ۷۵۰
نصف ستون (۱۷×۸) سانتیمتر " ۱۵۰۰
یک ستون کامل (۳۵×۸) سانتیمتر " ۳۰۰۰
نصف دو ستون (۱۷×۱۷) سانتیمتر " ۳۰۰۰
دو ستون کامل (۳۵×۱۷) سانتیمتر " ۶۰۰۰
نصف صفحه (۳۵×۲۶) سانتیمتر " ۴۵۰۰
تمام صفحه (۳۵×۲۶) سانتیمتر " ۹۰۰۰

برای درج در صفحه آخر، ۲۰ درصد به مبلغ فوق اضافه می‌شود.

شرایط پذیرش

آگهی باید به صورت آماده برای چاپ حداکثر تا سومین چهارشنبه ماه قبل از انتشار به دفتر مجله برسد و محتوای آن متناسب با شأن یک مجله علمی باشد. گزارش کامپیوتر در قبول یا رد آگهیها آزاد است



گزارش تشکیل جلسه مجمع عمومی

به دنبال رسمیت یافتن جلسه مجمع عمومی انجمن در دعوت اول که خبر آن در شماره گذشته ماهنامه به اطلاع رسید، جلسه دوم مجمع عمومی در روز چهارشنبه پانزدهم آبان ماه ۱۳۶۴ در محل سالن کنفرانس وزارت فرهنگ و آموزش عالی برگزار گردید. پس از رسمیت یافتن جلسه (بر طبق اساسنامه انجمن، هرگاه جلسه مجمع عمومی در دعوت اول به دلیل عدم حضور بیش از نیمی از اعضای عادی انجمن رسمیت نیابد، در دعوت دوم، جلسه مجمع عمومی با هر تعداد رسمی خواهد بود)، ابتدا از سوی دبیر انجمن مواد دستور جلسه قرائت گردید و سپس در مورد تک تک این مواد به شرح زیر، بحث و رای گیری به عمل آمد:

۱- افزودن چند ماده جدید به اساسنامه

اولین بند از دستور جلسه به افزودن مواد به اساسنامه انجمن اختصاص داشت. اخیراً از سوی وزارت کشور و شهرهای جمهوری اسلامی مقرر گردیده است که مواذیر به اساسنامه تمامی انجمنهای علمی افزوده شود و در غیر این صورت، از صدور پروانه وادامه فعالیت آنها جلوگیری به عمل خواهد آمد. این مواذیر قرار زیر می باشد:

الف: در صورتی که تصمیم به انحلال انجمن گرفته شود یا به عللی انجمن منحل گردد کلیه دارائیهای انجمن اعم از منقول یا غیر منقول (پس از وصول مطالبات و پرداخت بدهیهای) با نظروالی فقیه به یکی از مواذیر سات خبریه داخل کشور و گذار خواهد شد.

ب: پس دفترو شعب فرعی انجمن پس از کسب موافقت شهرهای جمهوری اسلامی قابل اجرا خواهد بود.

پ: انتشار هرگونه مطبوعه ای پس از اجازه از وزارت ارشاد اسلامی و دریافت پروانه انتشار با رعایت کامل قانون مطبوعات انجام خواهد شد.

پس از بحث و تبادل نظر درباره موافق و یا توجه به لزوم افزودن آنها به اساسنامه جهت ادامه فعالیت های انجمن، با تصویب مجمع عمومی این مواذیر به اساسنامه انجمن افزوده شدند.

۲- گزارش مالی و تصویب تراژنامه

در دومین قسمت از دستور جلسه مجمع عمومی

گزارش مالی و تصویب تراژنامه قرار داشت. ابتدا گزارش مالی سال گذشته انجمن که در شماره قبل ماهنامه به چاپ رسیده بود بین حضار توزیع شد. سپس از سوی افراد حاضر سوالاتی در مورد این گزارش به عمل آمد که توسط خزاندهار انجمن بدانها پاسخ گفته شد. سرانجام، تراژنامه فعالیت های مالی سال گذشته انجمن، به تصویب مجمع عمومی رسید.

۳- افزایش حق عضویتها

سومین بند از دستور جلسه مجمع عمومی به افزایش حق عضویتها برای سال ۶۶-۶۵ اختصاص داشت. ابتدا در مورد افزایش قابل توجه هزینه های انجمن و عدم امکان ادامه فعالیتها جاری با توجه به میزان موجود حق عضویتها، توضیحاتی ارائه گردید. همچنین نتیجه نظرخواهی از اعضاء از طریق پرسشنامه نیز به اطلاع مجمع عمومی رسانده شد. سپس، با توجه به این که حق عضویت اعضاء تنها منبع درآمد مستمر انجمن می باشد و نیز با توجه به عدم ثبات سایر درآمدهای انجمن از قبیل کمکهای مالی افراد و سازمانهای مختلف و شهریه کلاسهای آموزشی، پیشنهادیهای تاجرائی به منظور تأمین هزینه های انجمن از طریق افزایش حق عضویتها به مجمع عمومی ارائه گشت که با تغییرات مختصر به شرح زیر به تصویب رسید.

حق عضویت اعضای عادی ۲۵۰۰ ریال
حق عضویت اعضای وابسته ۲۵۰۰ ریال
حق عضویت اعضای دانشجویی ۱۰۰۰ ریال
حق عضویت اعضای حقوقی حداقل ۱۵۰۰۰ ریال
اشتراک ماهنامه برای یکسال ۲۰۰۰ ریال

لازم به تذکر است که میزان جدید حق عضویتها مربوط به دوره عضویت ۶۶-۶۵ است و از شهریور سال آینده قابل اجرا خواهد بود.

۴- انتخابات هیأت تاجرائی

در چهارمین بخش از دستور جلسه مجمع عمومی برگزاری انتخابات هیأت تاجرائی قرار داشت. ابتدا آراء رسیده از نظر تکراری نبودن وداشتن علامت در کنار نام بیش از ۲ نفر، مورد بررسی قرار گرفتند و پس از اراحت آنها، قسمت پرفراژ شده کناری برگه های رای که حاوی شماره عضویتها بود از آنها جدا گردید و قرائت آراء به عمل آمد. جمعا ۱۲۷ رای حضورا و ۱۵۱ رای به وسیله پست به انجمن واصل گشته بود که دو رای از آنها به علت جدا شدن پرفراژ کنساری باطل شناخته شد و از مجموع ۱۲۵ رای خوانده شده، افراد زیر برای عضویت در چهارمین هیأت تاجرائی انجمن برگزیده شدند:

- ۱- آقای ابراهیم نقیب زاده مشایخ ۱۱۱ رای
- ۲- خانم لیلی حاجی اسماعیلی ۱۰۸
- ۳- آقای حمیدرضا رهبر ۹۳
- ۴- آقای محمد میرزا عبداللهی ۸۹
- ۵- آقای مهدی فلاحتی ۸۱

- ۶- آقای عبداللہ کسرائی ۷۶
- ۷- آقای داریوش جوان ۵۹
- ۸- آقای محسن رستمی ۵۹
- ۹- آقای مهرداد مدبرنیا ۵۹

به دلیل تساوی آراء سه نفر آخر، قرار شد در اولین جلسه هیأت تاجرائی جدید، در مورد انتخاب یکی از آنها به عضویت هیأت تاجرائی تصمیم گیری لازم به عمل آید. بهر حال، یکی از نفرات فوق به عنوان عضو اصلی و دو نفر دیگر به عنوان اعضای البدل خواهند بود. آراء سایر نندیدها به عضویت در هیأت تاجرائی به قرار زیر بود: آقای مهرداد فیهمی (۵۱ رای)، خانم میترا آل باسین (۳۳ رای)، آقای میر مرتضی عبدی اسکوشی (۱۰ رای)، و آقای کبیا نوش خستوشی (۵ رای).

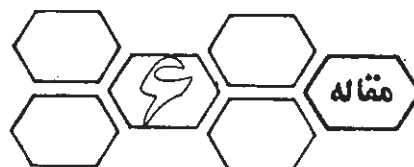
۵- بحث آزاد درباره فعالیت های آتی انجمن

در بحث آزاد که همزمان با بررسی وقرائت آراء و پس از آن صورت گرفت، موارد زیر از سوی شرکت کنندگان در مجمع عمومی مطرح گشت. این موارد در دستور کار هیأت تاجرائی جدید قرار خواهند گرفت:

الف: پیشنهاد شد حق عضویت در انجمن و حق اشتراک ماهنامه از یکدیگر جدا شوند و هر عضو در صورت تمایل به اشتراک ماهنامه گزارش کمیسیون، حق اشتراک آن را جداگانه بپردازد. این پیشنهاد به منظور تقویت بنیه مالی انجمن مطرح گردید.

ب: در مورد محل کتابخانه انجمن و امکان دستیابی به مستندات موجود در آن سوال شد. در این مورد توضیح داده شد که به علت نبودن محل مناسب برای کتابخانه کتابها و مستندات به صورت بسته بندی شده نگهداری می شوند و امکان دستیابی واستفاده از آنها نیست و طبعا به همین دلیل، تلاشی نیز از سوی مسئولین هیأت تاجرائی برای به هنگام درآوردن موجودی کتابخانه و تهیه کتاب و مجلات جدیدتر صورت نگرفته است.

پ: پیشنهاد شد جهت تقویت بنیه مالی انجمن و تأسیس دفتر، با ارگانهای ذیربط کشور نظیر شورای عالی انفورماتیک و وزارت فرهنگ و آموزش عالی تماس برقرار گردد. در این مورد نیز توضیح داده شد که تا حدی که استقلال انجمن محفوظ بماند، تماسهای گرفته شده است و لیست متقاضیان اکنون جواب مساعدی دریافت نشده است و تنها در سال گذشته، کمکهای محدودی از سوی وزارت فرهنگ و آموزش عالی به انجمن به عمل آمده که روی این گونه کمکها نیز نمیتوان به عنوان درآمد مستمری برای انجمن حساب کرد.



شناختی با پیشانیان علم کامپیوتر

۱۷

جان ویلیام ماکلی^۱
سازنده اولین کامپیوتررقمی

ترجمه: * ابراهیم نقیب زاده مشایخ

جان ماکلی، در ۳۰ اوت ۱۹۰۷ در شهر سینسیناتی اوها یوبه دنیا آمد و در ۸ نوامبر سال ۱۹۸۰ در شهر فیلادلفیا دیده از جهان فرو بست. او یکی از پیشانیان عمده عصر کامپیوترهای الکترونیک ورقمی امروز ما به شمار می آید. کامپیوتر ENIAC که محصول فکر توانای وی بود، در سال ۱۹۴۶، دنیای علم و محیط پردازش اطلاعات را به کلی دگرگون ساخت. در سال ۱۹۲۵، جان ماکلی به دنبال دریافت یک بورس تحصیلی، وارد مدرسه مهندسی دانشگاه جان هاپکینز شد. او پس از ۲ سال تحصیل در این مدرسه، دریافت که علاقه چندانی به این رشته ندارد و لذا به رشته فیزیک روی آورد. در سال ۱۹۲۲، موفق به اخذ دکترای فیزیک گشت. رساله دکترای وی در مورد تجزیه و تحلیل مولکول متواکسیدکربن بود. او به مدت یک سال دیگر به عنوان دستیار تحقیقاتی پیروفسور "ژوزف ایکوس" در دانشگاه جان هاپکینز باقی ماند و در خلال این مدت، بر روی محاسبه سطوح انرژی طیف فرمالدئیدها^۲ به کار پرداخت. این پروژه تحقیقاتی، همانند موضوع رساله دکترای ماکلی، دربرگیرنده انجام محاسبات بسیار زیادی بود و به همین دلیل، علاقه وی به یافتن روشهای خاصی جهت کاستن از حجم محاسبات، جلب گشت.

دکتر ماکلی از سال ۱۹۳۳ تا ۱۹۴۱ در کالج "اورسینوس"^۳ به تدریس فیزیک اشتغال داشت. در خلال این دوره، علاقه وی به مساله پیش بینی وضع هوا جلب شد و جهت تجزیه و تحلیل داده های هواشناسی، به ساختن یک کامپیوتر قیاسی^۵ پرداخت. این کار، موضوع مقاله ای شد که توسط او در سال ۱۹۴۰ انتشار یافت. جان ماکلی اوقات کاری خود را در تابستان سال ۱۹۴۰ با دکتر "هلم کلیتون"^۶ که علاقه تحقیقاتی وی در زمینه پیش بینی بلندمدت وضع هوا بود، صرف نمود. در طول این مدت، مقاله ای نیز تحت عنوان "بررسی آماری علل وجود نقاط تاریک بر روی خورشید"، منتشر ساخت.

در تابستان سال ۱۹۴۱ و به هنگامی که تهدیدات شروع جنگ مرتباً افزایش می یافت، او در یک دوره آموزش الکترونیک در مدرسه

مهندسی "مور"^۷ در دانشگاه پنسیلوانیا شرکت جست. سپس از وی دعوت به عمل آمد تا به عنوان مدرس به هیأت علمی مدرسه مزبور بپیوندد. این مدرسه، قرارداد طولی المدتتی با اداره تدارکات ارتش آمریکا برای محاسبه جداول بالستیک داشت و به ماکلی، علاوه بر وظایف آموزشی، این کار نیز محول گردید. حجم محاسباتی این وظیفه جدید، به تنهایی برابر با همه کارهای محاسباتی او در دهه گذشته بود. در سال ۱۹۴۲، پیشنهاد ساخت یک ماشین حساب الکترونیکی جهت انجام این محاسبات حیاتی را ارائه داد. این پیشنهاد دیدار مورد قبول واقع نشد اما یک سال بعد با تجدید نظری که به عمل آمد، مورد موافقت قرار گرفت. سی ماه بعد، در فوریه سال ۱۹۴۶، ENIAC^۸ زاده فکرتوانای جان ماکلی و ساخته شده زیر نظر مهندس "پرسپر اکرت"^۹، به نمایش عمومی گذاشته شد. این ماشین از نقطه نظر سرعت محاسباتی، جهش فوق العاده ای به حساب می آمد و



توانا شبهای محاسباتی را تا ۱۰۰۰ برابر سریع تر انجام میداد.

ENIAC که اولین کامپیوتر خودکار و همه منظوره^{۱۰} الکترونیک بود، به مدت ۱۰ سال بطور موفقیت آمیزی کار کرد. در سال ۱۹۴۶، "ماکلی" و "اکرت" مدرسه "مور" را ترک گفتند و شرکت "کنترل الکترونیک"^{۱۱} را پایه گذاری نمودند. اولین قراردادهای این شرکت جدیداً لتاء سیس، طراح یکی کامپیوتر دودویی^{۱۲} کوچک (BINAC) برای صنایع هواپیما سازی "نورتروپ"^{۱۳} و کامپیوتر دیگری به نام UNIVAC I برای اداره شرماری بود. در سال ۱۹۵۱، شرکت "ماکلی" و "اکرت"، به عنوان بخشی از شرکت بزرگ "زمینگتون-رند"^{۱۴} درآمد ماکلی تا سال ۱۹۵۹ که شرکتی به نام خودتاء سیس کرد، در آنجا باقی ماند. جان ماکلی یکی از بنیانگذاران انجمن ماشینهای محاسب (ACM)، اولین نایب رئیس، و دومین رئیس آن بود. وی همچنین

درجوامع علمی دیگری نیز از قبیل انجمن فیزیک آمریکا، انستیتوی فرانکلین، و آکادمی ملی مهندسان آمریکا عضویت داشت. جان ماکلی در دوران حیات علمی پربارش به جوایز متعددی دست یافت که از آن میان میتوان به جایزه "هاوارد پاتز"^{۱۵} انستیتوی فرانکلین (۱۹۴۹)، جایزه "جان اسکات"^{۱۶} (۱۹۶۱)، و جایزه "مردسال فیلادلفیا" (به همراه "اکرت" در سال ۱۹۷۳) اشاره کرد. همچنین، انجمن ماشینهای محاسب (ACM) به افتخار خدمات علمی ارزنده "ماکلی" و "اکرت"، نام آن دو را بر یکی از جوایز معتبر خود نهادهاست. مقالات علمی ماکلی، تنها در کتابخانه دانشگاه پنسیلوانیا نگاهداری میشوند.

با دداشتهای و مرجع

- 1: Encyclopedia of Computer Science and Engineering, VNR, 1983.
- 2: John William Mauchly
- 3: Joseph Eachus
- 4: Formaldehyde
- 5: Ursinus
- 6: Helm Clayton
- 7: Moore
- 8: Electronic Numerical Integrator and Computer
- 9: Presper Eckert
- 10: General-purpose
- 11: Electronic Control Co.
- 12: Binary Computer
- 13: Northrop
- 14: Remington-Rand
- 15: Howard Potts
- 16: John Scott Award

آگهی شماره ۷۰/۱

استخدام

شرکت ایران ارقام وابسته به وزارت برنام و بودجه از دانش آموختگان رشته کامپیوتر یا کسانی که سابقه کار و تجربه مفید در زمینه کامپیوتر داشته و علاقمند به فعالیت در زمینه بازاریابی و امور مشتریان هستند، دعوت به همکاری مینماید.

از متقاضیان تقاضا میشود مشخصات و شرح حال علمی و تجربی خود را به صورت دستی تهیه و با ذکر آدرس دقیق و شماره تلفن به صندوق پستی ۱۳۱۵ - ۱۴۱۵۵ ارسال نمایند.

به مراجعات حضوری و تلفنی پاسخ داده نخواهد شد.



پایداشت

همان گونه که اطلاع دارید، تعدادی از اعضای وابسته و دیگر خوانندگان گزارش کامپیوتر درخواست کرده اند که از طریق صفحه "پاسخ به سؤالات"، مطلبی به زبان ساده در مورد مفاهیم اولیه و اجزاء تشکیل دهنده کامپیوترها (و به ویژه کامپیوترهای شخصی) درج گردد. در پاسخ به این تقاضا، تاکنون سه بخش از مقاله حاضر در شماره های پیش به نظرتان رسیده است و اینک توجهتان را به چهارمین و آخرین قسمت از این مقاله جلب میکنیم.

گامی به سوی کامپیوترهای شخصی

(قسمت آخر)

ترجمه: * محمدحسن برادران قاسمی

حافظه کامپیوتر عموماً به دو طبقه تقسیم میشود: یکی به نام حافظه کاری (که غالباً به نام RAM استفاده کننده خوانده میشود) و دیگری ROM از پیش برنامهریزی شده که حافظه انبوه نامیده میشود.

حافظه کاری را میتوان به صورت مجموعه ای از چند "خانه" در نظر گرفت که هر یک از آنها قابلیت ذخیره فقط و فقط یک بایت اطلاعات را دارند. هر یک از این خانه ها یک نشانی منحصربه فرد دارند. خانه ها هر یک به مجموعه ای از "نقاط" تقسیم شده اند که مجموع آنها برابر با تعداد بیت هایی که طول کلمه کامپیوتر را تشکیل داده اند، میباشد. بطور مثال، در یک کامپیوتر ۸ بیتی، هر خانه حافظه دارای ۸ نقطه است که میتواند رقم یک و صفر را که تشکیل دهنده طول کلمه اش میباشد، در خود جای دهد. هر نقطه در یک خانه حافظه، فقط یک عدد بیک یا صفر را میتواند دربرداشته باشد. هرگز دو مکان حافظه نمیتوانند دارای نشانی یکسانی باشند ولی درونشان ممکن است محتلف، میتواند یک علامت، حرف، یا عدم مشابهی قرار داشته باشد.

وقتی که قسمت پردازنده کامپیوتر به یک واحد اطلاعاتی از حافظه احتیاج داشته باشد، نشانی آن محل را گرفته و مستقیماً به سراغ آن جهت اخذ داده های مورد نظر خود می رود. پردازنده برای این کار، نیازی به گام برداشتن متوالی در خانه های حافظه ندارد. بلکه مستقیماً به محلی که میخواهد برود، "پرش" مینماید. این قابلیت مراجعه مستقیم به هر محل مورد نظر، به پردازنده کامپیوتر اجازه دستیابی تصادفی

به نشانی خواسته شده را میدهد. بدین دلیل، ترتیب محل نشانیها به نام حافظه یا دستیابی تصادفی یا RAM خوانده میشود.

در اکثر کامپیوترها، بیشتر حافظه های کاری به صورت RAM هستند ولی بخشی از حافظه کاری نیز برای نوع دیگری از حافظه تخصصی می باشد که به آن، حافظه فقط خواندنی یا ROM گفته میشود. مشابه نوع RAM، در نوع ROM نیز دستیابی به نشانیها میتواند به صورت تصادفی باشد. ولی از آنجائی که ROM حاوی برنامه ای است که در یک توالی خاص باید مورد پردازش قرار گیرد، دستیابی به محلهای آن معمولاً به صورت تصادفی نیست. بنا بر این، حافظه فقط خواندنی به عنوان حافظه بدستایابی تصادفی شناخته نمیشود.

یک اختلاف اساسی بین RAM و ROM این است که RAM جهت ذخیره موقت داده ها تا زمانی



که به آنها احتیاج باشد به کار میرود، در حالی که ROM حاوی داده هایی است که به صورت دائمی در آن ذخیره شده اند. داده هایی که در RAM قرار دارند، قابل پاک شدن و جایگزین شدن با داده های دیگر هستند ولی داده هایی که در ROM قرار گرفته اند، بطور عادی توسط استفاده کننده کامپیوتر قابل پاک شدن نیستند. بنا بر این، RAM را باید تخته سیاهی پنداشت که میتوان با رها و بارها مطالب مختلفی را بر روی آن نوشت و پاک کرد ولی ROM، مشابه یک کاغذ تایپ شده است که قابل پاک شدن نبوده و همیشه حاوی اطلاعات ثابتی است.

یک اختلاف اساسی دیگر بین RAM و ROM مربوط به شکل حضور داده ها در این دو نوع مختلف است. داده های ذخیره شده در RAM تا پایدار هستند، به این مفهوم که فقط تا زمانی که حافظه باقی میماند که توان مصرفی حافظه (برق) تا مین شده و وقفه ای در آن ایجاد نشده باشد

و هر نوع وقفه در توان مصرفی RAM موجب از بین رفتن اطلاعات موجود در آن میگردد و زمانی که مجدداً توان مصرفی تاءمین گردد، حافظه RAM کامپیوتر حاوی هیچگونه اطلاعاتی نخواهد بود. از طرف دیگر، ROM پایدار میباشد بدین مفهوم که کلیه اطلاعات موجود در آن، همانطور که کارخانه سازنده در آن نهاده است، صرف نظر از این که توان مصرفی آن تاءمین شده یا نشده باشد، در آن باقی خواهد ماند.

RAM و ROM هر دو دستگا های الکترونیکی و در واقع، مشابه ریزپردازنده ها، تراشه های مدار مجتمع میباشند. این دستگا ها به همراه قسمتهای پردازش و ورودی و خروجی، بخشی را تشکیل میدهند که به آن سخت افزار کامپیوتر گفته میشود (به ROM به عنوان بخش نرم افزار هم میتوان نگاه کرد، چون حاوی برنامه های لازم جهت عملکرد کامپیوتر میباشد. در واقع به دلیل آن که ROM از پیش برنامهریزی شده، بسیاری از خصوصیات نرم افزاری سخت افزاری را تاءمین می دهد، غالباً "FIRMWARE" خوانده میشود).

حافظه انبوه

حافظه انبوه یک قدم از پردازنده گذشته کامپیوتر دورتر است و اطلاعات آن فقط از طریق عبور از حافظه کاری میتواند در اختیار پردازنده قرار گیرد.

ظرفیت زیاد حافظه های انبوه، روش نسبتاً کم هزینه ای را جهت مقابله با نیاز کامپیوتر به ذخیره حجم عظیمی از داده ها در اختیار میگذارد. حافظه انبوه، معمولاً مرکب از یک ماده مغناطیسی است که میتواند اطلاعات مورد نیاز را در خود ضبط نماید. دو وسیله اساسی به عنوان دستگا حافظه انبوه معمولاً در کامپیوترهای شخصی مورد استفاده واقع میشوند. این دو عبارتند از: کاست و دیسک.

کاست از رایج ترین نوع حافظه انبوه است (بین ۵۰ تا ۱۵۰ دلار). هرچند که این وسیله حجم زیادی از داده ها را میتواند در خود جای دهد ولی جهت انجام محاسبات جدی زیاد به کار نمیرسد. در مقایسه با دیسک، حافظه کاست نسبتاً نامطمئن، کند، و فاقد انعطاف پذیری لازم است. علت ذکر کردن آن در اینجا این بود که با دآر و شوم چنین چیزی وجود دارد و به عنوان وسیله ای ابتدائی با قیمت کم میتواند مورد استفاده واقع شود.

قدم بعدی دروسایل حافظه انبوه چسبی است که به نام "دیسک لرن" خوانده میشود. این وسیله همچنانکه از اسمش برمیآید، به فرم صفحات قابل انعطاف است که لایه ای از ترکیبات مغناطیسی بر هر دو طرف آن کشیده شده است و این لایه، قسمت ضبط کننده را تشکیل میدهد. بسته به نوع گرداننده دیسک و نوع دیسکی که استفاده میشود، یک یا هر دو طرف قابلیت نوشتن و ورق برزیدن



جهت دست یافتن به ظرفیت ذخیره بسیار بالاتر، نیازمند یکا رگیری دیسکهای سخت می‌باشیم. ظرفیت این دیسکها از ۵ میلیون بایت تا متجاوزا ۱۶۰ میلیون بایت اطلاعات است. دستگاه حافظه داخل گرداننده‌های دیسک سخت (که غالباً "گرداننده‌های وینچستر" خوانده می‌شوند) شامل صفحات فلزی محکم یا سختی هستند که بر روی آنها پوششی از ماده ضبط کننده (اکسید منگناطیسی) داده شده است. خود صفحات داخل محفظه ضدغباری محفوظ شده‌اند. نیاز به حافظه انبوه‌تر با یک مسأله اقتصادی است. حافظه‌های کاری سریع و بلافاصله دسترسی پذیر، نه فقط به دلیل گرانی خود دستگاهها بلکه به دلیل فضای که در داخل کامپیوتر می‌گیرند و توان مصرفی مورد نیازشان گران تمام می‌شوند و لازم ضرورت استفاده از حافظه انبوه را مطرح می‌سازند.

پاداشتها و منعی

- 1: Random Access Memory
- 2: Preprogrammed Read Only Memory
- 3: Mass Memory
- 4: Random
- 5: Volatile
- 6: Integrated-circuit Chips
- 7: Floppy disk
- 8: Hard Disks

• Getting Started in Personal Computing, Alexander W. Burawa, Computer & Electronics, Jan. 1984.

ذخیره برنا مهودا دهه را دارد. اسم "لرزان" به خاطر ماهیت انعطاف پذیر دیسک است و هرگاه که دیسک مورد استفاده واقع نمی‌شود، درلفافی پلاستیکی قرار می‌گیرد.

دیسکهای لرزان در اندازه‌های مختلف عرضه شده اند که بسته به قطر آنها از ۸ اینچ تا ۱/۲ اینچ به نام دیسکهای کوچک، و ۳ اینچ تا ۴ اینچ به نام دیسک ریز، خوانده می‌شوند. هر یک از این اندازه‌ها برای گرداننده‌های خاصی طراحی شده اند و باید یکدیگر را سازگار می‌باشند. بر حسب ظرفیت ذخیره سازی، دیسک ۸ اینچی به میزان قابل توجهی بیشتر از حافظه دیسک ۱/۲ اینچی ظرفیت دارد ولی ظرفیت دیسکهای ریز معادل دیسکهای ۱/۲ اینچی می‌باشد. دیسکهای لرزان، قابلیت ذخیره اطلاعات از ۸۰،۰۰۰ بایت (بطور خلاصه ۸۰K) در انواع قدیمی تر گرداننده‌های ۱/۲ اینچی تا بالای ۱ میلیون بایت (بطور خلاصه ۱M) در جدیدترین انواع گرداننده‌ها را دارند. هستند که این عمدتاً بستگی به یک یا دوری بودن صفحات منگناطیسی و گرداننده‌های آنها دارد. به عنوان یک قاعده، نوع دوری، ظرفیتی معادل دوبرابر یا بیشتر از انواع یک رویه دارد.

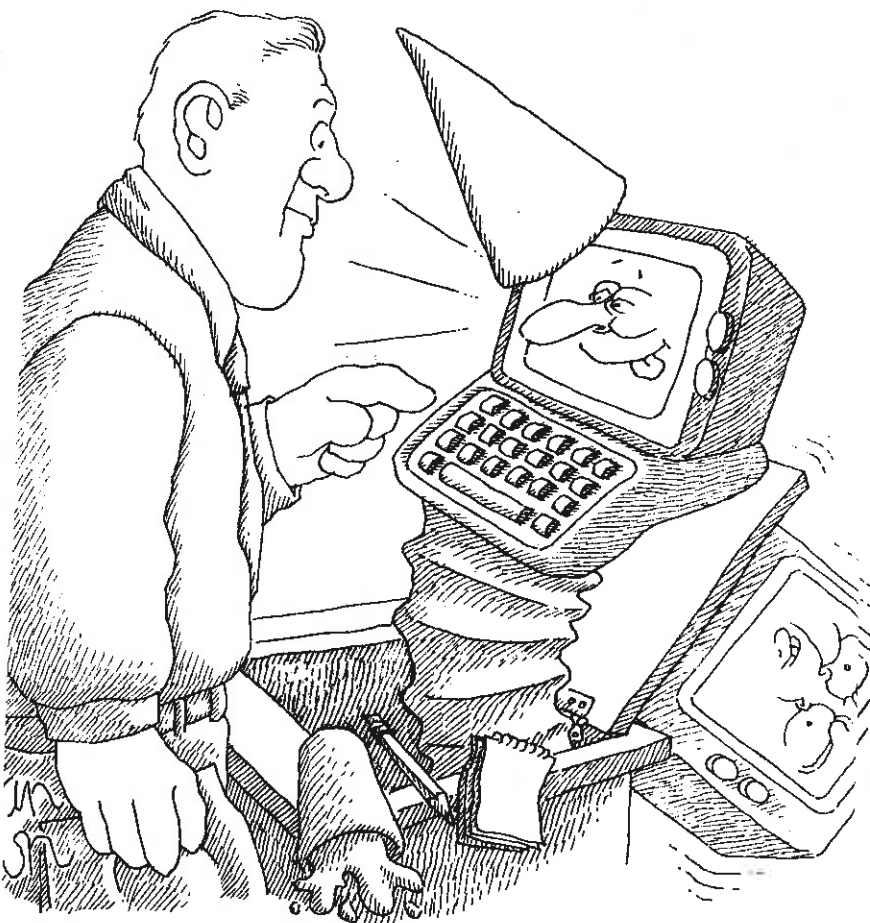


فرصت طلبی در بازار کامپیوتر

شرکتهائی که محصولات خود را به بازارهای ناپایدار عرضه می‌کنند، گاه به زیانهای هنگفتی مواجه می‌شوند. اما شرکت "لوگوس کامپیوتر-سیستمز"، که همواره به دنبال بازارهای پرسود بوده است، در این مورد خلاصه نگران نیست. در حوالی سال ۱۹۶۹، شرکت لوگوس در زمینه طراحی سیستم کامپیوتری برای ترجمه خودکار از انگلیسی به ویتنامی فعالیت می‌کرد. ارتش آمریکا که برای ترجمه دستورالعملها و جزوه‌های راهنمای تجهیزات نظامی به این سیستم نیاز داشت، یکی از مشتریان مهم شرکت لوگوس بود. خروج عجلانده آمریکا از ویتنام، شرکت لوگوس را از یک مشتری عمده محروم ساخت. اما شرکت لوگوس چند هزار کیلو متر به سوی غرب آمدواران زمان شاه را به عنوان مشتری پولدار بعدی در نظر گرفت و به طراحی سیستم ترجمه خودکار انگلیسی به فارسی برای وزارت دفاع مشغول شد. این منبع درآمد خیلی زود با وقوع انقلاب از بین رفت. شرکت لوگوس که از شر خوبی ندیده بود، با زحمات به طرف غرب رفت و با مشترک اروپا را به عنوان هدف بعدی خود برگزید. در حال حاضر، چنین به نظر می‌رسد که این بازار پابدارتر از دوزیا رقبلی باشد، اما کسی چه می‌داند؟

ترجمه آزاد: میروزیهای

از: کامپیوتینگ، اکتبر ۱۹۸۵





اعضاء

تغییرات در آمار اعضای انجمن

افراد زیر تا تاریخ ۱۳۶۴/۸/۱۵ به عضویت انجمن انفورما تیک ایران درآمده اند:

۱۶۴۰	میشر	امان الله	ع
۱۶۴۱	ژند	مهرداد	ع
۱۶۴۲	رشیدفرخی	عیسی	ع
۱۶۴۳	کتا پستان	رکسانا	ع
۱۶۴۴	کتا نچی	بهروز	ع
۱۶۴۵	سلوکی طبالبودانی	ناصر	ع
۱۶۴۶	روبرت	چوردن	ع
۱۶۴۷	بحران	محمدکاظم	ع
۱۶۴۸	زند	امیرابراهیم	ع
۳۱۹۱	بهرامی	فرزین	و
۳۱۹۲	بهرامیان	خلیل	و
۳۱۹۳	میثاقیان	امیر	و
۳۱۹۴	خوارزمی	شهین دخت	و
۳۱۹۵	حسانی	امیرمسعود	و
۵۵۳۲	مزارعی	فرزاد	د
۵۵۳۳	فرمان	سعید	د
۵۵۳۴	توتونچی افشار	هما یون	د
۵۵۳۵	شعیبی	شهریار	د
۵۵۳۶	فرجی	بهرام	د
۵۵۳۷	صفا شانی	مهرداد	د
۵۵۳۸	امامی فرد	بهزیر	د
۵۵۳۹	فاضلی	سیدکاظم	د
۵۵۴۰	جواد گم	شهرام	د
۵۵۴۱	حجتی	سیدابوالحسن	د
۵۵۴۲	براق جمالی	جواد	د
۵۵۴۳	بافکر	آرتا	د
۵۵۴۴	عمادی	محمدجعفر	د
۵۵۴۵	طاهری	محمد	د

تجدید عضویتها

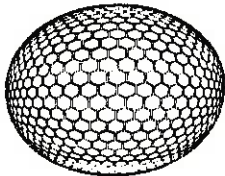
افراد زیر تا تاریخ ۱۳۶۴/۸/۱۵ عضویت خود را در انجمن تمدید نموده اند:

۵۲۵۹	شرکت معصوم	مرتضی	د
۵۲۸۴	هشتروزی زاد	شاهین	د
۵۲۳۵	انصاری	زهرا	د
۱۳۸۷	بنای یزدی پور	احمد	ع
۵۳۷۰	هخامنشی	مهران	د
۱۴۲۴	غفاریان	احمد (۶۳-۶۴)	ع
۳۰۸۴	رجبعلی پور	سیدی (۶۳-۶۴)	و
۵۱۷۸	طیبری رضاشی	هما	د
۵۴۳۳	سینائی خسروشاهی	سیده آسیه	د
۵۳۵۹	اوحدی لاهیجانی	فرامرز	د
۳۰۸۲	جیل عا ملی	فرشید	و
۱۳۱۲	حق نگهدار	سیروس	ع
۵۲۱۵	پاشا مقدم	احمد (۶۳-۶۴)	د
۵۴۳۲	پورفرد	فرحناز	د
۵۴۳۴	سیرجانی	مرجان	د
۳۱۳۹	زریپاک	شاهین	و
۵۲۶۹	وقفی مجرد	حسین (۶۳-۶۴)	د
۱۵۰۲	ورمی	کاظم	ع
۱۴۳۵	پناه هی	سید غلامرضا	ع
۱۵۲۷	هیوی	ابوالقاسم	ع
۱۶۰۳	قلعی	علی اکبر	ع
۱۰۵۶	تاسری	پرویز	ع
۳۱۵۷	کشیشیان	مانوک	و
۵۳۹۲	رسولی	بهروز	د
۵۰۰۰	مالچی	موش	د
۵۰۶۲	افروز	زهرا	د
۵۴۳۷	هانی طباشی	سیدحسین	د
۱۵۷۹	شاکری فرد	منصور (۶۳-۶۴)	ع
۱۵۵۴	گودرزی	سیاوش	د
۵۴۳۰	قدوس	پریسا	د
۱۱۱۶	برهانی	بهشید	ع
۱۶۳۹	زندنی	حسین (۶۳-۶۴) ازبده	ع

۵۲۸۸	باقری	قدرت	د
۵۳۹۰	ذوالفقاریان	مهرداد	د
۵۴۳۸	گلزاریان	کامران	د
۱۵۶۴	مقبلی	امیراحمد	ع
۱۶۱۶	شیخ هرنیدی	خسرو	ع
۵۲۹۱	جلیلی	رسول (۶۳-۶۴)	د
۵۳۸۲	کیانی	محمد رضا	د
۵۱۸۲	گلشن امیری	آتیلا	د
۵۱۶۴	مدیقی	سیدمحسن	د
۱۴۳۸	فرخی	محمد	ع
۵۰۵۵	قاضی سعیدی	مصطفی	د
۱۳۰۰	فلاحی	مهدی	ع
۳۱۱۰	دیانندی	ژاکلین	و
۳۱۱۵	مزارعی	فریبرز	و
۵۳۹۴	ابهری	عبدالله	د
۱۳۰۳	نهرودی	علی اکبر	ع
۱۵۶۳	شهری طبهرستانی	علی	ع
۱۵۱۲	فاضل پور	علیرضا	ع
۱۵۸۱	اقبال	نادر	ع
۳۱۱۴	فلاحی	حسین	و
۵۱۷۰	نصیری زاده	فریدون	د
۱۵۷۳	مرعشی	جعفر (۶۳-۶۴)	ع
۲۱۴۴	میلانی	مسعود	و
۳۱۶۳	دارائی	هااله	و
۳۱۴۷	فردبهبهانی	امیرمنصور	و
۱۶۲۷	چمنی آذر	حمید	ع
۵۳۳۵	شنتیانی	مهران	د
۵۴۰۴	تکا پوی	ناهد	د
۵۴۰۳	محمودیان مقدم	طیبه	د
۵۴۱۰	حشمتی	قدس سینا	د
۵۴۶۶	پریت سیان	هامو	د
۵۴۹۵	حبیبی	خسار	د
۵۴۶۴	کتانی	کامبیز	د
۵۴۶۲	آزما	فریبرز	د
۱۴۸۵	شیروانی	عباس	ع
۵۴۳۱	قدوسی	محمد رضا	د
۵۴۰۶	سعادتمندی	سعیده	د
۵۴۴۶	امینی علویجه	داود	د
۵۴۱۸	شهابی	سیروس	د
۵۴۲۰	طغرل	نادر	د
۵۴۰۸	نسیمی	ارژنگ	د
۵۲۰۵	عیسوی چنینه	رومینا (۶۳-۶۴)	د

بدا بین ترتیب آمار اعضای انجمن تا تاریخ ۶۴/۸/۱۵ به قرار زیر است:

عادی	وابسته	دانشجویی	برجسته	حقوقی	جمع
۳۰۲	۱۰۶	۳۶۹	۱	۷	۷۸۵



۵ مهره زردیف اول یا ۳ مهره زردیف دوم ، بازی را به حالت "ا من" تبدیل نماید :

۱	۲	۴	۸	۱	۲	۴	۸
۱	۰	۰	۰	۱	۰	۰	۰
۱	۰	۰	۰	۱	۰	۰	۰
۱	۰	۰	۰	۱	۰	۰	۰
۱	۰	۰	۰	۱	۰	۰	۰

این عمل "ا من" سازی وضعیت در واقع یک عمل منطقی است که در نزد برنامه نویسان زبان اسمبلر به نام XOR (بای انحصاری) موسوم می باشد. آنچه XOR انجام می دهد این است که بیت های موجود در دو عدد را مقایسه کرده و در هر جا که بیت های آن یکسان نباشد مقدار یک و در غیر این صورت مقدار صفر قرار می دهد. به بیان دیگر :

$$\begin{aligned} 0 \text{ XOR } 1 &= 1 \\ 1 \text{ XOR } 0 &= 1 \\ 1 \text{ XOR } 1 &= 0 \\ 0 \text{ XOR } 0 &= 0 \end{aligned}$$

زبان بیسیک کامپیوتر شما احتمالاً فاقد دستور XOR است ولی شما می توانید به کمک دستورهای AND ، OR ، و NOT به ترتیب زیر آن را ایجاد نمایید :

$$A \text{ XOR } B = (A \text{ OR } B) \text{ AND } (\text{NOT}(A \text{ AND } B))$$

اینک می توانید برنامه مهره بازی را بر روی کامپیوتر خود پیاده کنید و از دوستان خود دعوت کنید تا با برنامه مهره هوشمند شما به رقابت بپردازند.

گردآوری و ترجمه
از : دکتر پروینا صر

- 1: NIM
- 2: Charles Leonard Bouton
- 3: Exclusive OR

۳ - هروضعیت "نا ا من" را می توان با یک بازی مناسب به وضعیت "ا من" تبدیل نمود. همانطور که ملاحظه می شود، کافی است که شخصی خود را در یک وضعیت "ا من" قرار بدهد. طرف مقابل سپس فقط می تواند آن وضعیت را به هم بزند و شخص می باید مجدداً سعی نماید تا وضعیت "ا من" جدیدی ایجاد نماید.

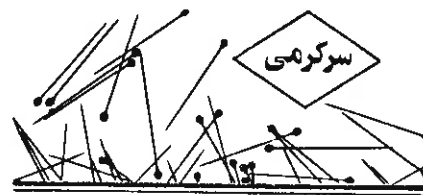
برای تعیین وضعیت بازی، تجزیه و تحلیل دودویی ارقام بازی ضروری است. به مثال زیر توجه کنید. فرض کنید در یک حالت بازی با مهره ۱۳، ۱۲، ۴ باشد. ابتدا با اید این ارقام را به صورت دودویی در زیرهم نوشت. خواهیم داشت :

۱	۲	۴	۸
۱	۰	۱	۱
۰	۰	۱	۱
۰	۰	۱	۰

سپس تعداد یک ها را در هر ستون شمرده و اگر تعدادشان زوج باشد، در زیر آن ستون عدد صفر و در غیر این صورت، عدد یک قرار می دهیم :

۱	۲	۴	۸
۱	۰	۱	۱
۰	۰	۱	۱
۰	۰	۱	۰
۱	۰	۱	۰

اگر آنچه که در زیر جدول ایجاد می شود، تماماً صفر باشد، وضعیت "ا من" قلمداد می گردد و در غیر این صورت، "نا ا من" است. اینک با زیرین دیگر می باید از یکی از ردیف های جدول بسه تعداد مناسب عدد کم کنند تا وضعیت جدول به "ا من" مبدل گردد و به بیان دیگر، با انجام عمل فوق الذکر، چیزی جز صفر در زیر جدول قرار نگیرد. برای مثال، وی می تواند با برداشتن



بازی "نیم"

بازی "نیم" یک بازی فکری قدیمی است که در آن، دو بازیکن تعدادی مهره را در سه ردیف با بیشتر قرار داده و سپس یکی از آنها یک ردیف را انتخاب کرده و به هر تعداد که می خواهد، آن را از آن ردیف ها برداشته و آن را در یک ردیف (یا حتی تمام مهره های آن ردیف) را کنار می گذارد و نوبت را به دیگری واگذار می کند. بازیکن دیگری نیز به همین رویه به بازی ادامه می دهد. نفر برنده کسی است که آخرین مهره را از میان بردارد.

استراتژی این بازی را یک استاد ریاضی دانشگاه ها رو ادبه نام "چارلز لئون بوتا" در سال ۱۹۰۱ کشف کرد. سادگی این بازی از یکسو و استراتژی آن از سوی دیگر باعث گردید که این بازی جزء اولین آزمایشاتی باشد که در هوش مصنوعی بکار می رود. به بیان دیگر، برنامه نویسان سعی کردند نحوه این بازی را به کامپیوتر یاد بدهند. در زیر، استراتژی این بازی را بحث کرده و سپس برنامه ای را معرفی می کنیم که به کمک آن کامپیوتر شخصی شما بجای یک بازیکن قرار می گیرد و به بازی می پردازد. استراتژی این بازی به شرح زیر است :

۱ - هروضعیت بازی فقط به یکی از دو حالت "ا من" یا "نا ا من" است.

۲ - اگر وضعیت بازی به حالت "ا من" باشد، هر حرکت طرف مقابل آن را به وضعیت "نا ا من" تبدیل می کند.

READY.

```
10 SCNLCL:U$="NO"
12 INPUT "ENTER NUMBER OF ROWS (3-18)";R
14 IF R<3 OR R>18 THEN PRINT
    "MUST BE BETWEEN 3 AND 18":GOTO 12
16 DIM RO(R)
18 FOR I=1 TO R
20 PRINT "ENTER NUMBER OF OBJECTS (1-25)
    IN ROW";I:INPUT RO(I)
22 IF RO(I)<1 OR RO(I)>25 THEN PRINT
    "MUST BE BETWEEN 1 AND 25":GOTO 20
24 NEXT:GOSUB 62
26 INPUT "ENTER 1 TO PLAY FIRST, 2 TO PLA
    Y SECOND ";M:IF M<1 OR M>2 THEN 26
28 ON M GOTO 30,46
30 P$="YOU"
32 INPUT "YOUR TURN. WHICH ROW";RO
34 IF RO<1 OR RO>R THEN PRINT
    "INVALID ROW":GOTO 30
36 INPUT "HOW MANY";RE
38 IF RE>RO(RO) OR RE<1 THEN PRINT
    "INVALID":GOTO 32
40 RO(RO)=RO(RO)-RE:GOSUB 62
42 IF U$="YES" THEN END
44 PRINT "HIT ANY KEY":GETKEY A$
46 P$="I":P=RO(1)
48 FOR I=2 TO R
```

```
50 P={P OR RO(I)}AND(NOT(P AND RO(I)))
52 NEXT
54 IF P=0 THEN GOSUB 78:ELSE GOSUB 88
56 GOSUB 62
58 PRINT "I HAVE MOVED";RE; "FROM ROW";RO
60 IF U$="NO" THEN 30:ELSE END
62 SCNLCL:S=0
64 FOR I=1 TO R
66 PRINT "ROW";I; " ":RO(I); " ":TAB(15);
68 IF RO(I)=0 THEN S=S+1:GOTO 72
70 FOR J=1 TO RO(I):PRINT "X";NEXT
72 PRINT:NEXT
74 IF S=R THEN PRINT P$; " WON":U$="YES"
76 RETURN
78 RE=0:RO=0
80 DO WHILE RE=0:RO=RO+1
82 IF RO(RO)>0 THEN RE=1
84 LOOP
86 RO(RO)=RO(RO)-RE:RETURN
88 RE=0:RO=0
90 DO WHILE RE=0:RO=RO+1
92 RT=(RO(RO)OR P)AND(NOT(RO(RO)AND P))
94 IF RT<RO(RO) THEN RE=RO(RO)-RT
96 LOOP
98 RO(RO)=RO(RO)-RE:RETURN
```

READY.