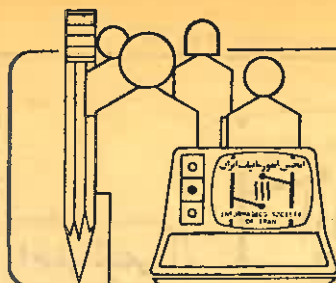


سال هشتم - شماره ۵
شماره پیاپی ۷۹
شهریور و مهر ۱۳۶۵



گزارش کامپیوتر

ماهنامه انجمن انفورماتیک ایران





اطلاعیه

برندگان جایزه خوارزمی سال ۱۳۶۴

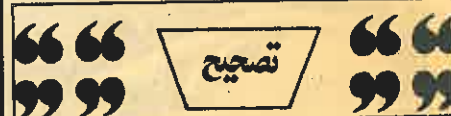
هیئت داوران برگزیده هیئت اجرایی، پس از مطالعات، بررسیها و بحث و تبادل نظر، بهترین مقالات چاپ شده در ماهنامه های گزارش کا مپیوتر سال ۱۳۶۴ را معرفی نمود.

معیارها و ضوابط در نظر گرفته شده جهت ارزیابی مقالات عبارت بودند از:

- روانی مطلب (۲۵ امتیاز)
- جالب بودن و تازگی موضوع مقاله (۲۰ امتیاز)
- مطلوبیت متن ارائه شده از لحاظ وجود زیر نویس، نمودارها و شکلها لازم، کامل بودن مراجع و غیره (۲۰ امتیاز)
- پیچیدگی موضوع و متن مقاله اصلی در مورد ترجمه، کیفیت و صحت علمی متن در صورت تألیف (یا تأکید بیشتر بر روی تألیف به جای ترجمه) (۲۵ امتیاز)
- میزان کار انجام شده و طول مقاله (۱۰ امتیاز)

هیئت داوران، پس از بررسی مقدماتی، ۱۱ مقاله از بین ۶۸ مقاله چاپ شده در سال ۶۴ را جهت شرکت در مسابقه برگزید و بر حسب معیارها و ضوابط فوق، آنها را مورد ارزیابی قرار داد. البته بنا به خواست اعضای هیئت داوران و برای رعایت کامل بیطرفی، مقالات ترجمه و یا تألیف شده توسط این اعضاء در مسابقه شرکت داده نشدند.

بقیه در صفحه ۲۱



ضمن پوزش از خوانندگان عزیز توجه آنان را به تصحیحات زیر مربوط به شماره گذشته جلب می‌کنیم:

بر روی جلد نشریه تیرومرد ۶۵ (پیاپی ۷۸)، به اشتباه درج شده بود: "سال هشتم - شماره ۳ که صحیح آن "سال هشتم - شماره ۴" می‌باشد.

در ضمن در مرحله تهیه فیلم و زینت هنگام مرمت و رتوش فیلمها، شماره صندوق پستی انجمن در قسمت "برای تماس" از صورت صحیح خود خارج و همچنین کلمه "در" از یکی از جملات برجسته مقاله شماره ۱ پاک گردیده بود.



اخبار انجمن

قرار رسیدن موعد تمدید عضویتها

با شروع مهرماه، موعد تمدید عضویت اعضای انجمن فرا رسید و کلیه اعضا می‌بایست عضویت خود را برای سال ۶۵-۶۶ تمدید نمایند. اطلاعات مربوط به مبلغ حق عضویت در صفحه "اعضای همین شماره" نشریه درج شده است. انجمن انفورماتیک ایران همه ساله تعدادی از اعضای خود را برای شرکتها در تمدید عضویت از دست می‌دهد.

انتظار می‌رود که کلیه اعضا، با توجه به بحران مالی انجمن، هر چه سریعتر به این مهم اقدام نمایند.

برگزاری دعوت اول مجمع عمومی انجمن

دعوت اول مجمع عمومی سالانه انجمن در تاریخ ۸ مهر ۱۳۶۵ در شرکت داده پردازی

ایران تشکیل شد که به علت عدم حضور اکثریت اعضای عادی رسمیت نیافت.

در این جلسه ضمن معرفی فعالیتهای کمیته‌های مختلف انجمن در سال گذشته، ترازنامه مالی انجمن ارائه گشت اما به علت رسمی نبودن جلسه تصویب آن به جلسه رسمی مجمع عمومی در دعوت دوم موکول گشت. آراء نامزدهای انتخابات میان دورهای هیئت اجرایی نیز قرائت گردید و اعضای اصلی و علی‌البدل برای هیئت اجرایی انتخاب شدند.

مجمع عمومی در دعوت دوم رأس ساعت ۴/۳۰ روز ۲۷ آبان ۱۳۶۵ در محل وزارت فرهنگ و آموزش عالی واقع در خیابان استادیار تاج‌الهی برگزار خواهد شد.

انتخابات میان دورهای هیئت اجرایی

در دعوت اول هشتمین مجمع عمومی انجمن، انتخابات میان دورهای هیئت اجرایی به منظور تکمیل تعداد اعضای این هیئت، برگزار گردید.

در این رابطه ابتدا آرا کتبی اعضا جهت اطمینان از صحت آنها مطابق ضوابط مقرر (هر فرد بیش از یک رأی نداده و بیش از چهار نفر را انتخاب نکرده باشد) مورد بررسی بقیه در صفحه ۲۱

برای تماس با انجمن از نشانی یا شماره تلفنهای زیر استفاده کنید:

نشانی: انجمن انفورماتیک ایران، صندوق پستی ۱۱۹۶-۱۴۱۵۵ تهران

برای تماس

شماره تلفنهای مسئولان انجمن

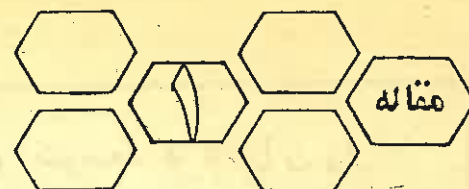
رئیس آقای ابراهیم نقیب‌زاده مشایخ داخلی ۲۴۷ و ۲۲۹۰۷۱-۲
 نایب رئیس آقای محمد میرزا عبداللهی
 دبیر خانم لیلی حاجی اسماعیلی ۸۹۱۲۹۳ و ۸۹۱۴۲۹
 خزانه‌دار آقای دکتر محسن رستمی ۸۹۱۲۹۳ و ۸۹۱۴۲۹

شماره تلفنهای سرپرستان کمیته‌ها

انتشارات آقای حمیدرضا رهبر ۸۹۱۲۹۳ و ۸۹۱۴۲۹
 آموزش آقای دکتر محسن رستمی ۸۹۱۲۹۳ و ۸۹۱۴۲۹
 فعالیتهای علمی و فرهنگی آقای محمد میرزا عبداللهی ۴۳۷۲۲۴۷
 روابط عمومی آقای ابراهیم نقیب‌زاده مشایخ داخلی ۲۴۷ و ۲۲۹۰۷۱-۲

دیگراعضای هیأت اجرایی

عضو اصلی آقای بهزار روحانی ۸۹۴۱۵۰
 عضو اصلی آقای حمید خورشید رحیم زاده ۲۱۴۱۶۱۲
 عضو علی‌البدل خانم میترا آل یاسین ۸۹۱۴۲۹ و ۸۹۱۲۹۳
 عضو علی‌البدل خانم میترا کا تو زیان ۶۱۳۹۳۳۳



ترجمه: بهنام اسراری

آدا : زبان برنامه نویسی نوین



نرم افزار/پردازش داده ها/ زبانهای برنامه نویسی/ زبان آدا /

فورترن و پاسکال مدت زیادی است که توسط مهندسين برای حل مسائل مسرود استفاده قرار می گیرند. اکنون انتظار می رود که زبان برنامه نویسی جدیدی به نام آدا مورد استفاده اکثر مهندسين، بخصوص افراد استخدای شده در کمپانیها بی که برای وزارت دفاع آمریکا (DOD) کار می کنند، قرار گیرد.

DOD استفاده کننده عمده سیستمهای نرم افزار و سخت افزار کامپیوتر می باشد. همان طور که می توان انتظار داشت سیستمهای بزرگ دفاع هوایی، زمینی و دریایی به سیستمهای محاسبه پیچیده ای نیاز دارند. بسیاری از این سیستمهای بزرگ، از قبیل کامپیوتر هدایت کننده موشک، در واقع از چندین زیر سیستم کوچک ساخته شده اند. هر کدام از این زیر سیستمهای جاسازی شده شامل مجموعه ای از کامپیوترها و نرم افزارهاست، و به نوبه خود می توانند دارای زیر سیستمهایی باشند. جهت کارآیی صحیح تمام سیستم، سیستمهای جاسازی شده باید با یکدیگر در ارتباط باشند. بنابراین یک سیستم بزرگ می تواند از چندین زیر سیستم توسعه یافته توسط کمپانیهای مختلف ساخته شده باشد. اگر موضوع را کاملاً ملتبس کنیم، سیستمهای جاسازی شده می توانند به زبان فورترن، اسمبلی، پاسکال نوشته شوند، و این منجر به سیستمهای عظیم و گرانبهای می شود که پیاده سازی و نگهداری آنها مشکل است.

DOD در تلاش برای کنترل هزینه های نرم افزار دفاعی، ضمن نگهداری قابلیت اطمینان سیستم، بانی توسعه زبان برنامه نویسی جدیدی به نام "آدا" شده است. آدا ضمن این که چندین خصیصه موجود در زبانهای فورترن و پاسکال را دارا می باشد، خاصیتی

◀

DOD زبان برنامه نویسی جدید و با ارزشی را توسعه داد و آن را به افتخار آدا لاولیس، اولین برنامه نویس جهان، آدا نامگذاری کرد.

اضافی دارد که آن را برای استفاده در سیستمهای بزرگ و پیچیده مهندسی نرم افزار ساده و مطمئن تر می سازد. در حال حاضر، DOD ملزم ساخته است که چندین سیستم نرم افزار جدید دفاعی به زبان آدا نوشته شوند. جهت معرفی آدا، تاریخ توسعه آن و ساختارهای اولیه این زبان را بررسی می کنیم. در اینجا فلسفه ای که در پشت طرز احسی برنامه های آدا وجود دارد، توسط طراحی و بهرمز در آوردن یک برنامه ساده که دو ماتریس را با یکدیگر جمع می کند مرور خوا ه شد. نکته آخر که به اندازه نکات فوق اهمیت دارد بررسی چشم اندازهای آینده این زبان در رابطه با کاربردهای دفاعی و غیر دفاعی است.

تاریخچه آدا

در اوایل سال ۱۹۷۰ وزارت دفاع ایالات متحده متوجه روندا افزایش هزینه های نرم افزاری و قابلیت اطمینان ضعیف برای سیستمهای عمده دفاعی شد. در سال ۱۹۷۵، DOD گروهی را تحت عنوان HOLWG^۱ جهت مطالعه نیازهای زبانهای برنامه نویسی DOD بوجود آورد. این گروه، زبانهای موجود در مقابل این نیازها را ارزیابی کرد و مجموعه ای از زبانهای برنامه نویسی مورد قبول را برای استفاده در سیستمهای دفاعی DOD توصیه نمود. در این کار علاوه بر کارشناسان DOD، عده ای از متخصصین صنایع و دانشگاهها شرکت داشتند. پیشنهاد نهایی HOLWG توسعه زبان کامپیوتری جدیدی بود که برای طراحی، پیاده سازی و نگهداری تمام جنبه های نرم افزار سیستمهای دفاعی مورد استفاده قرار گیرد. این گروه به علت نبود نام بهتری، زبان مورد نظر را DOD-1 نامیدند.

طراحی چنین زبان جدیدی کار کوچکی نبود. جهت شناختن این مسئله، DOD یک مسابقه طراحی بین المللی که در آن چندین تیم طرحهایی برای DOD-1 ارائه دادند برگزار کرد. طرح برنده توسط یک تیم فرانسوی از هانی ول - هانی ول بال توسط

ژان ایچیه^۵ ارائه شد.

تصمیم گرفته شد که نام DOD-1 را به افتخار آگوستا آدا با یرون^۶، کنست لاولیس و دختر لر دبا یرون شاعر، به آدا تغییر دهند. آدا لاولیس (۱۸۱۵-۱۸۵۱) ریاضیاتی بود که با چارلز بابیج بر روی ماشینهای تحلیلی و تفاضلی وی، که دو کامپیوتر مکانیکی بودند، کار می کرد. او (آدا) برنامه نویسی ماشینهای بابیج را پیشنهاد کرد. با این کار، وی به عنوان اولین برنامه نویس دنیا شناخته می شود.

به دنبال طرح، زمان آزمایشش و ارزیابی شروع شد. این مرحله شامل صدور دعوتی عمومی برای استفاده زکا ر بورد موجود و پیاپی کردن آن به زبان آدا بود. به علاوه، یک کتاب مرجع مقدماتی برای این زبان تهیه شد و توسعه یک همگردان (کامپایلر) آغاز گشت.

با توسعه همگردانهای آدا توسط تک شرکتها، توانایی آنها جهت دربرگرفتن استانداردهای DOD برای آدا، آزمایش شد و برای همگردانهای حاضر ایستگاهها استانداردها را رعایت می کردند. DOD سپس درخواست نمود که سیستمهای نرم افزار دفاعی با استفاده از همگردانهای معتبر آدا برای اجرا نوشته شوند. به این طریق، سیستمهای نرم افزاری که بر روی نوعی سخت افزار بنا شده بر روی نوعی سخت افزار دیگر ترکیب شوند. چنین سیستمهای نرم افزاری را سیستمهای قابل حمل^۸ گویند.

سرگذشت آدا هنوز کامل نیست. در حالی که تا کنون چند شرکت چون دیتا جنرال^۹ دیجیتال ایکوئیتیمنت (DEC)^{۱۰}، و تل - سافت^{۱۱} همگردانهای آدای مورد تأیید DOD را گسترش داده اند همگردانهای دیگری بزودی در دسترس خواهند بود. کار همگردان بر روی توسعه ابزار نرم افزاری برای استفاده در نوشتن، اشکال زدائی و نگهداری برنامه های آدا ادامه دارد. این ابزار محیط پشتیبانی برنامه نویسی آدا (APSE)^{۱۲}



واحد برنامه	مشخصه	موارد استفاده
زیربرنامه - رویه - تابع - بسته - وظیفه	عمل ترتیبی مجموعه منابع پرداختهای موازی	واحد برنامه اصلی انجام عملیات برگرداندن هیچ، یک یا چند ارزش تعاریف زیربرنامه‌های گروهی انجام عملیاتی که می‌توانند به طور همزمان اتفاق بیافتند

جدول ۱: واحدهای برنامه آدا

در این شکل، خط اول نوعی به نام `COUNTER - TYPE` را معرفی می‌کند که عدد صحیح است. خط دوم یک مفهوم (متغیر) را تعریف می‌کند که برای نگهداشتن مقدار (ارزشی) از نوع `COUNTER - TYPE` (یعنی عدد صحیح) می‌باشد. خط سوم یک نوع اعشاری متمیز را تعریف می‌کند که دقتش ۱۵ رقم است. سپس مفهوم `ORBIT - LENGTH` تعریف می‌شود که از نوع `PLANETARY - MEASUREMENT` است. خط پنجم تعریف `MATRIX - TYPE` است که آرایه‌ای دوبعدی از اعداد صحیح می‌باشد. توجه کنید که زیرنویسها، اعداد صحیح مثبت هستند ولی خط پنجم هیچ حوزه‌ای از مقادیر برای این زیرنویسها نمی‌دهد. وقتیکه ما تریسهای `A` و `B` در خط ششم تعریف می‌شوند محدوده زیرنویسها مشخص می‌شوند. بنا بر این `A` و `B` ما تریسهایی از مقادیر صحیح هستند که هر کدام دوسطر و سه ستون دارند.

`MOTOR - STATE` به عنوان نوعی شمارنده در نظر گرفته می‌شود. مفاهیم `MOTOR - STATE` می‌توانند یکی از سه ارزش: `OFF`, `FORWARD` یا `REVERSE` را داشته باشند. در خط هشتم `HOIST` از نوع `MOTOR - STATE` تعریف و مقدار اولیه `OFF` به آن داده می‌شود.

بعد از تعریف مفاهیم فوق می‌توانیم به بدنه برنامه آدا وارد شویم، جایی که واقعا "می‌توانیم این مفاهیم را دستکاری کنیم.

آدا: زبان و ابزار طراحی

آدا علاوه بر این که زبان برنامه نویسی است ابزاری نیز برای طراحی سیستمها می‌باشد. این روش برخورد با مسئله طراحی را "مفهوم‌گرا" می‌نامیم که پیشرفتگی در زمینه اسلوب شناسایی موجود در طراحی برنامه به شما می‌رساند زیرا مدل سازی سیستمهای واقعی را ساده تر کرده، روشی جهت طراحی سیستمهای در واقع خود مستند ساز را ارائه



شوند. این عمل در قسمت مشخصات واحد انجام می‌شود. بدنه شامل رمزآدای بکار رفته برای انجام عملیات مورد نظر می‌باشد. شکل ۱ قسمتهای اساسی یک رویه واحد برنامه را نشان می‌دهد. نام رویه `MATRIX - DEMO` است. احکامی که با علامت - شروع می‌شوند توضیحات آرایه‌ای آدا می‌باشند.

```

procedure MATRIX_DEMO is
--The specification part goes
--in this section.
begin
--The body of the procedure
--goes in this section.
end MATRIX_DEMO

```

شکل ۱: ساختمان یک رویه آدا

برای تعریف کردن یک مفهوم در برنامه آدا با بدهم نام و هم نوعش را ذکر کنیم. مثالیابی از انواع مفهوم عبارتند از: صحیح، اعشاری (متمیزش و یا متمیزنا بت) و آرایه.

روش مرسوم این است که ابتدا خود نوعها را به طور جداگانه تعریف می‌نمایند و سپس بر مبنای آنها مفاهیم را تعریف می‌کنند. مفاهیم در فورترین و پاسکال متغیرنا می‌ده می‌شوند. به مثالهای نشان داده شده در شکل ۲ توجه کنید.

```

type COUNTER_TYPE is integer;
number of days: COUNTER_TYPE;

type PLANETARY_MEASUREMENT is digits (15);
orbit length: PLANETARY_MEASUREMENT;

type MATRIX_TYPE is array
  (POSITIVE range <1> ..
   POSITIVE range <2>) of integer;
A: MATRIX_TYPE (1, 2, 3, 3);

type MOTOR_STATE is (OFF, FORWARD, REVERSE);
hoist: MOTOR_STATE := OFF;

```

شکل ۲: مثالهایی از تعاریف نوع و مفهوم در آدا

را تشکیل می‌دهند. با گذشت زمان، این ابزارها مانند خود آدا استانداردهای می‌شوند به طوری که `APSE` موجود بر روی تمام انواع سخت افزارها یکسان خواهد شد.

یک مسئله بسیار مهم دیگر، آموزش برنامه نویسان آدا می‌باشد. از آنجایی که بیشتر مهندسين، با زبان فورترین و پاسکال آشنا هستند ولی با آدا آشنا نیستند، مجبور خواهند شد که این زبان جدید را یاد بگیرند. این معرکه برای مهندسين جوان طالب کار که هنگامت برای بسیاری از شرکتها می‌باشد. در این میان، مهندسين جوان طالب کار که با زبان آدا آشنا هستند جذب این شرکتها خواهند شد.

مروری بر ساختمان آدا

برنامه آدا از یک یا چند واحد برنامه می‌شود تشکیل شده است که هر کدام می‌توانند به طور جداگانه همگرا شوند. این واحدها زیربرنامه‌ها، بسته‌ها، رویه‌ها، تابعی، آرایه‌ای مشخصات و موارد استفاده مشابه با نظائر خود در فورترین و پاسکال می‌باشند (جدول ۱).

بسته‌ها روش مناسبی از دسته‌بندی زیربرنامه‌ها را فراهم می‌آورند. زیربرنامه‌ها می‌توانند با توجه به وظایفشان یا به دلیل اینکه اطلاعات مشترکی دارند دسته‌بندی شوند. به عنوان مثال گروهی از زیربرنامه‌ها می‌توانند به در طراحی و تجزیه و تحلیل مدار می‌توانند به صورت بسته واحدی درآیند. این مفهوم شبیه به کتابخانه‌های برنامه ۱۹ در فورترین و پاسکال است.

"وظایف" در سیستمهایی که پردازش در آنها به طور موازی صورت می‌گیرد مهم هستند. سیستمی را در نظر بگیرید که یک سری از دستگاهها را مبدی می‌کند، تجزیه و تحلیل داده‌های هر دستگاه را انجام می‌دهد، و نتایج را چاپ می‌نماید. این وظیفه در آن واحد چاپ جمع آوری اطلاعات از یک دستگاه، تجزیه و تحلیل آنها از یک دستگاه ثانوی، و چاپ نتایج را برای یک دستگاه سوم می‌دهد. گرچه این وظیفه در فورترین و پاسکال مشکل است ولی پیاده سازی آن در آدا آسان است و صرفاً "با توسعه یک واحد برنامه برای هر کدام از سه وظیفه صورت می‌گیرد. بقیه کار توسط آدا انجام می‌شود.

به طور کلی هر واحد برنامه آدا دو قسمت دارد: مشخصات^{۲۲} و بدنه^{۲۳}. برخلاف فورترین، ولی مانند پاسکال، تمام مفاهیم^{۲۴} (متغیرها و ثابتهای خاص) با بدنه آدا تعریف

نتایج نشان می دهند که آدا طراحی، پیاده سازی و نگهداری سیستمهای پیچیده و بزرگ نرم افزاری را ساده می کنند.



ما هرکدام از اینها را در مورد مسئله جمع ما تریس مورد بحث قرار می دهیم. مفاهیم مورد نظر، سه ما تریس هستند: A و B و RESULT. اینها ما تریسهای هستند که عناوینشان اعداد صحیح می باشند. عمل، جمع ما تریس است. ساختن رابطه اش را به ما بجا دقت مست تشخیص واقعی تابع دارد. قبل از توجه کردن به الگوریتمهای مورد استفاده در حل مسئله، ملاحظه می کنید که این کار در ما حل اولیه طرح مفهوم گرا صورت می گیرد. شکل ۲ نشان می دهد که چگونه این رابط می تواند تعریف شود.

```
Function MATRIX-SUM (A,B: MATRIX-TYPE)
return MATRIX-TYPE is
--declare RESULT
RESULT: MATRIX-TYPE
(A-RANGE(1),A-RANGE(2));
begin
--The function body will
be placed here later.
end MATRIX-SUM;
```

شکل ۳: مشخصات MATRIX - SUM

در این شکل، RESULT از نوع MATRIX - TYPE (با توجه به شکل ۲) تعریف شده است. حوزه سطرها و ستونهای RESULT باید با ما تریس A یکی باشد. ما تریسهای A و B همراه با MATRIX - TYPE در واقع در رویت های تعریف می شوند که MATRIX - SUM را فرا می خواند. خطوط پنجم و ششم در شکل ۲ نشان می دهند که چگونه این کار صورت می گیرد. MATRIX - SUM یک مقدار (ارزش) از نوع MATRIX - TYPE را بر می گرداند.

آخرین مرحله در طراحی این تابع آدا پیاده سازی آن می باشد، یعنی، نوشتن احکامی که بدنه تابع را تشکیل می دهند. شکل ۴ تابع تکمیل شده را نشان می دهد.

```
Function MATRIX-SUM (A,B: MATRIX-TYPE)
return MATRIX-TYPE is
--declare RESULT
RESULT: MATRIX-TYPE
(A-RANGE(1),A-RANGE(2));
begin
for I in A-RANGE(1) loop
for J in A-RANGE(1) loop
RESULT(I,J)
:= A(I,J) + B(I,J);
end loop;
end loop;
return RESULT;
end MATRIX-SUM;
```

شکل ۴: تابع MATRIX - SUM

می دهد، بدین معنی که طرح و توسعه یک سیستم به صورتی انجام می شود که نیاز به مستند سازی اضافی سیستمها به طور قابل ملاحظه ای کاهش می یابد. تحقیق کنونی روی سیستمهایی انجام شده است که می توانند طرحهای مفهوم گرا را گرفته، آنها را به برنامه های آدا تبدیل کنند.

مؤلفه های اساسی فن طراحی مفهوم گرا عبارتند از:

۱- تعریف مسئله.

۲- توسعه یک خط مشی غیر رسمی برای حل مسئله.

۳- فرموله کردن این خط مشی.

مؤلفه اخیر شامل تشخیص مفاهیم مورد نظر و عملکردشان، ساختن رابطه ایی ۲۷ میان واحدهای مختلف برنامه که مفاهیم را پردازش می کنند، سپس پیاده کردن اعمال می شود.

برای توضیح طرح مفهوم گرا و تهیه مقدمه ای بر زبان برنامه نویسی آدا، طرح و پیاده سازی یک تابع ساده را در نظر بگیرید که جمع دو ما تریس را انجام می دهد. اصول طراحی مفهوم گرا را دنبال می کنیم:

۱- تعریف مسئله:

بیمت یک تابع آدا که مجموع دو ما تریس را حساب کند و نتیجه را برگرداند.

۲- توسعه یک خط مشی غیر رسمی برای حل مسئله:

مجموع دو ما تریس A و B با اضافه کردن عناوین آنها در نظر گرفته می آید. برای مثال اگر نتیجه A+B در یک ما تریس به نام RESULT برگردانده شود، ما برای هر عنصر در A و B محاسبه می کنیم:

$$RESULT(I,J) = A(I,J) + B(I,J)$$

تعداد سطرها و ستونهای تمام ما تریسها باید با هم برابر باشد. فرض می کنیم تمام عناوین ما تریس اعداد صحیح هستند و این تابع توسط یک رویت فرا خوانده می شود.

۳- فرموله کردن خط مشی:

الف - تشخیص مفاهیم مورد نظر و عملکردشان.

ب - ساختن رابطه ایی در بین واحدهای برنامه.

ج - پیاده کردن اعمال.

بهما ختا رابطه توجه کنید. (1) A-RANGE به محدوده زیرنویسهای ستونی و (2) A-RANGE به محدوده زیرنویسهای ستونی ما تریس A اشاره می کنند. زیرنویسهای آدا می توانند اعداد صحیح، مثبت، صفر یا منفی باشند. مفاهیم طبقه، I و J تعریف نمی شوند و مقادیر آنها پس از خروج از حلقه های مربوط به خودشان، نامشخص هستند. بعد از خروج از حلقه بیرونی مقدار نتیجه به عنوان مقدار تابع برگردانده می شود.

اکنون این تابع می تواند توسط هر واحد آدا که نیاز به جمع کردن ما تریسها داشته باشد به کار برده شود. شکل ۵ رویه ساده ای را نشان می دهد که به ما تریسهای A و B مقدار اولیه داده، A+B را انجام داده، و نتایج را چاپ می کند.

IO - TEXT موجود در خطوط اول و دوم و نهم یک بسته آدا از رویه های ورودی/خروجی است. Put و new-line رویه های بی در IO - TEXT هستند.

در خط هشتم، به آدا گفته شده که تابع MATRIX - SUM در فایلی جداگانه قرار دارد. به جای گذاشتن MATRIX - SUM در یک فایلی جداگانه، می توان آن را در رویه جداگانه MATRIX - DEMO گنجاند. برای این منظور کافی است که خط هشتم را با احکام نشان داده شده در شکل ۴ جایگزین کنیم.

وقتی برنامه اجرا می شود دستور خط ۱۲، ما تریسهای A و B را به تابع MATRIX - SUM رد می کند. یکبار که ما تریسها اضافه می شوند

یادداشتها و مرجع :

- * Daniel C. St. Clair, "Ada: a new programming language", IEEE potentials, Oct. 1985, pp. 26-29.
1. Ada
 2. U.S. Department Of Defense
 3. Higher Order Language Working Group
 4. Honeywell/Honeywell Bull
 5. Jean Ichbiah
 6. Augusta Ada Byron
 7. Countess of Lovelace
 8. Transportable
 9. Data General
 10. Digital Equipment Corporation
 11. Telesoft
 12. Ada Programming Support Environment
 13. Program unit
 14. Subprograms
 15. Packages
 16. Tasks
 17. Procedural
 18. Functional
 19. Program Libraries
 20. Monitoring
 21. Specification
 22. Body
 23. Objects
 24. Comments
 25. Object-oriented
 26. Self-documenting Systems,
 27. Interfaces

طراحی، پیاده سازی و نگهداری سیستمهای نرم افزاری بزرگ می‌باشد. وزارت دفاع تنها نیازمند به چندین سیستم دفاعی جدید طرح شده و برنا مریزی شده با استفاده از آدا نیست، بلکه سازمان ملی هوانوردی فضایی (ناسا) در حال سرمایه گذاری در مورد استفاده از آدا برای بسیاری از سیستمهای است که قرار است در ایستگاه فضایی جدید آن که در اوایل سال ۱۹۹۰ مستقر می‌شود، مستورد استفاده قرار گیرند.

هرچند آدا جدید است و تعداد همگراهای تأیید شده DOD نسبتاً کم می‌باشد، نتایج نشان می‌دهند که آدا به راستی طراحی، پیاده سازی و نگهداری سیستمهای نرم افزاری پیچیده و بزرگ را ساده می‌کند.

قابلیت عمل رمز، توانمهای روش طراحی مفهوم گرا، باعث پیشرفت ارتباط و درک بین تمام افراد درگیر طراحی و پیاده سازی سیستمهای عظیم شده است.

انتظار می‌رود که استفاده از آدا در کاربردهای غیر دفاعی تا اندازه‌ای کندتر از کاربردهای دفاعی باشد. یکی از دلایل این امر کمبود فعلی برنامهنویسان آدا است. دلیل دیگر نبودن نرم افزار کمکی است که به آدا متصل شود. به عنوان مثال، برنامهای که به قابلیت‌های خاصی در رابطه با کاربردهای جدید نیاز دارند یا یک سیستم مدیریت بانک اطلاعاتی مجهز شود. با تکامل آدا این مشکلات از بین می‌روند و آدا مورد استفاده زیادی در زمینه‌های غیر دفاعی خواهد داشت.

در حالی که آدا نقطه پایانی بر تمام زبانهای برنامهنویسی نیست، اما یکی از بهترین ابزار موجود جهت برخورد با سیستمهای نرم افزاری پیچیده و بزرگ است. از این نظر می‌توان انتظار داشت که آدا برای مدت زمانی در آینده مورد استفاده قرار گیرد.

نتایج برگشت داده شده در ماتریس SUM قرار می‌گیرند. خروجی برنامها، ماتریس

۶	۸	۱۰
۱۲	۱۴	۱۶

است.

شکل ۵ همان برنامها که به پاسکال UCSD برای ماشین آپل نوشته شده نشان می‌دهد. در این شکل، زیربرنامها مربوط به جمع ماتریس در داخل برنامها اصلی گذاشته شده است. گرچه آپل پاسکال UCSD تا اندازه‌ای محدودتر از همگراهای دیگر پاسکال است، اما این برنامها مقایسه‌های را بین پاسکال و آدا ارائه می‌دهد.

```
with TEXT_IO;
use TEXT_IO;
procedure MATRIX_SUM is

  type MATRIX_TYPE is array
    (POSITIVE range <>, POSITIVE range <>)
    of INTEGER;

  A : MATRIX_TYPE (1..2, 1..3) :=
    ((1, 2, 3), (4, 5, 6));
  B : MATRIX_TYPE (1..2, 1..3) :=
    ((5, 6, 7), (8, 9, 10));
  SUM : MATRIX_TYPE (1..2, 1..3);

  function MATRIX_SUM (A, B : MATRIX_TYPE)
    return MATRIX_TYPE is separate;

package INT_IO is new
  TEXT_IO_INTEGER_IO (INTEGER);
use INT_IO;

begin

  SUM := MATRIX_SUM (A, B);

  for I in SUM.range(1) loop
    for J in SUM.range(2) loop
      put (SUM(I, J));
    end loop;
    new_line;
  end loop;
  new_line (2);
end MATRIX_SUM;
```

شکل ۵: رویه آدا که از SUM - MATRIX استفاده می‌کند.

آینده آدا

چندین خصیصه آدا مورد بحث قرار گرفته است. این خصایص از ادایک وسیله قوی در

بقیه از صفحه ۱۵

علاوه برکنگره‌های جهانی، کمیته‌های فنی مختلف این فدراسیون سالانه به برگزاری تعداد زیادی کنفرانس بین‌المللی در زمینه‌های تخصصی گوناگون اقدام می‌نمایند. مقالات ارائه شده در این کنفرانسها همگی چاپ شده و مخزن اطلاعات با ارزشی را راجع به تاریخ و روند توسعه علم پردازش اطلاعات و دانش‌های کنونی بشر در باره این علم تشکیل می‌دهند.

یادداشتها و مرجع :

- * ENCYCLOPEDIA OF COMPUTER SCIENCE AND ENGINEERING,

8. WORKING GROUPS (WG)

9. INTERFACES

10. DATA COMMUNICATION

11. LOCAL COMPUTER NETWORKS

12. INTERNATIONAL COMPUTER MESSAGE SYSTEMS

13. WORLD CONFERENCE ON MEDICAL INFORMATICS (MEDINFO)

14. INTERNATIONAL ASSOCIATION FOR MATHEMATICS AND COMPUTERS IN SIMULATION (IMACS)

15. INTERNATIONAL FEDERATION OF AUTOMATIC CONTROL (IFAC)

16. FIVE INTERNATIONAL ASSOCIATIONS COORDINATING COMMITTEE (FIACC)

VOL.1, PP797-799

I.L. AUERBACH

1. INTERNATIONAL FEDERATION FOR INFORMATION PROCESSING (IFIP)

2. INTERNATIONAL CONFERENCE ON INFORMATION PROCESSING

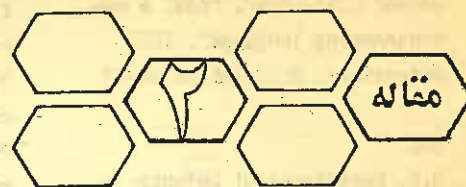
3. IFIP APPLIED INFORMATION PROCESSING GROUP (IAG)

4. INTERNATIONAL MEDICAL INFORMATICS ASSOCIATION (IMIA)

5. INTERNATIONAL ASSOCIATION FOR PATTERN RECOGNITION (IAPR)

6. INTERNATIONAL ASSOCIATION FOR STATISTICAL COMPUTING (IASC)

7. TECHNICAL COMMITTEES (TC)



مقدمه‌ای بر زبان لیسپ



ترجمه و گردآوری: مبتدیان یاسین

هرچند سبک قواعد نحوی
لیسپ غیرمتعارف است
اما در واقع این زبان
یکی از ساده‌ترین زبانهای
برنامه‌سازی است.

نرم‌افزار/زبانهای برنامه‌نویسی/پردازش
لیسپ/زبان لیسپ/

لیسپ یکی از معروفترین زبانهای
پردازش لیستهاست. این زبان توسط
جان مک‌کارتی^۱ و همکارانش در اوایل دهه
۱۹۵۰ و اوایل دهه ۱۹۶۰ در دانشگاه ام‌آی‌تی
ایده‌آ شده.

برای آشنایی با زبان لیسپ، ابتدا
با بدیهه‌دو مفهوم "عبارتهای نماد" و
"عبارتهای فوق‌زبان" آشنا شویم.
عبارتهای نمادی^۲ (عبارتهای S): داده‌های
ورودی در زبان لیسپ، عبارتهای نمادی
نامیده می‌شوند. یک عبارت نمادی، میتواند
یک اتم یا یک لیست باشد.

هر اتم، یک رشته‌ی حروف و اعداد، نویسه‌ای^۳،
مشمول بر حروف بزرگ و ارقام است که با یک
حرف آغاز می‌گردد. ابتدا و انتهای هر اتم،
توسط پرانتزهای چپ و راست یا توسط علامت
فاصله خالی^۴ مشخص می‌گردد. اعداد نیز جزء

برنامه‌های زبان لیسپ با استفاده از
عبارتهای فوق‌زبان نوشته می‌شوند. زبان
نوشتن برنامه‌ها در لیسپ، در حقیقت
زیرمجموعه‌ای از زبان داده‌هاست. به
همین دلیل نوشتن برنامه‌هایی به زبان
لیسپ که بر روی برنامه‌های دیگر لیسپ عمل
می‌کنند (مانند کامپایلرها، بهینه‌سازها،
و مفسرها) بسیار ساده‌است.

زبان لیسپ در بین برنامه‌سازان به
عنوان یک زبان دشوار و پیچیده‌اشتهار
دارد. این پندارنا درست از سبک غیر
متعارف قواعد نحوی^۵ این زبان ناشی
می‌گردد. در حقیقت، زبان لیسپ یکی از
ساده‌ترین زبانهای برنامه‌سازی است.

واحد اساسی یک برنامه در زبان لیسپ
فرم^۶ نامیده می‌شود. یک فرم معمولاً تابعی
با نشانه‌های^{۱۰} مربوطه است. تقریباً
تمام فرمهای لیسپ از قالب زیرپنوی
می‌کنند:

(Function arg1 arg2 arg3 ...)
انواع دیگر فرمها عبارتند از: متغیرها و



اتمهای زبان لیسپ هستند. در زبان لیسپ،
اعداد صحیح و اعداد اعشاری با ممیز ثابت و
شناور وجود دارند. تمام توابع لیسپ که روی
اعداد عمل می‌کنند، به‌طور خودکار نوع عدد
را می‌آزمایند و تبدیلات لازم را انجام می‌دهند.
هر لیست از یک پرانتز چپ، تعدادی
اتم یا لیست، و یک پرانتز راست تشکیل
می‌شود.

به عنوان مثال هر یک از اقلام زیر یک
عبارت نمادی است:

DOG
1984
(WHERE IS TURING NOW)
((MC CARTHY) IS MASTER OF (THE DARK
TOWER))

{{(LIST STRUCTURE (((CAN)))) BE
{(((((VERY)))) ((DEEP))))}}

کلمه NIL یکی از اتمهای خاص در
زبان لیسپ است و برای مشخص ساختن
انتهای لیست به کار می‌رود. NIL به
تنهایی به معنای لیست تهی^۷ است.

عبارتهای فوق‌زبان^۸ (عبارتهای M):

- tions", Clark, K. and D. Cowell, MC Graw Hill, 1976
1. Lisp
 2. John Mc Carthy
 3. Symbolic Expressions
 4. Character
 5. Blank
 6. Meta Language Expressions
 7. Optimizers
 8. Syntax
 9. Form
 10. Argument
 11. Contents Of Decrement Part Of Register
 12. Contents Of The Address Part Of Register
 13. Predicate

بقیه از صفحه ۱۰

از آن استفاده نمود. با استفاده از زبان طراحی برنامه‌آدا، برنامه‌نویس می‌تواند توضیح واقعی طراحی خواهد بود و می‌توان اطمینان داشت که بین این دو یکپارچگی لازم برقرار است.

● محیط پشتیبان برنامه‌نویسی آدا (APSE): زبان آدا توسط گسترده‌ترین مجموعه ابزار برنامه‌نویسی پشتیبانی خواهد شد. با توجه به قابلیت حمل^۱ این زبان (وابسته نبودن به یک سخت‌افزار خاص) و این امر که ابزار فوق‌العاده‌ای به خود زبان آدا نوشته خواهند شد، باید قبول داشت که بهترین اجزاء و برنامه‌های کمکی APSE بر روی بسیاری از کامپیوترها قابل دسترس خواهد بود.

یادداشتها و مراجع

- * How to Succeed with ADA
Tony Orme, data processing,
June 1986
- 1. ADA
- 2. ADA Programming Support environment
- 3. John Barnes
- 4. Programming in the Large
- 5. modifiable
- 6. reliable
- 7. understandable
- 8. realtime
- 9. Program Design Language
- 10. Portability

بنا بر این:

(LAMBDA (x Y) CONS (Y X)) (A (A B C))
= (A A B C)

مفسر ابتدا نشانه‌ها را ارزیابی می‌کند. سپس این مقادیر به متغیرهای مربوطه در لیست متغیرهای لاندا نسبت داده می‌شوند. در مثال بالا، X دارای مقدار (A B C) و Y دارای مقدار A خواهد بود. بنا بر این (CONS Y X) دارای مقدار (A A B C) خواهد بود که این مقدار تمام فرم لانداست. در زبان لیسپ، گزاره‌هایی وجود دارد که با اعمال بر روی متغیرها، یکی از دو مقدار درست یا نادرست را می‌دهند. به عنوان مثال، ATOM یک گزاره تک‌نشانده‌ی است. اگر A یک اتم باشد، (ATOM A) = T و (ATOM(A A)) = NIL است. گزاره EQ، تساوی دو عنصر اتمی را مورد بررسی قرار می‌دهد:

(EQ A A) = T

(EQ A B) = NIL

گزاره‌ها به ویژه در فرمهای شرطی به‌کار می‌روند. فرم زیر نمونه‌ای از یک فرم شرطی است:

(COND ((ATOM X) X) (T (CAR X)))

اگر (A B C) = X باشد، مقدار فرم شرطی فوق A است.

هر یک از نشانه‌های یک فرم شرطی به صورت یک دوتایی مرتب هستند. نخستین عنصر دوتایی، گزاره "و دومین عنصر آن "مقدار" است. قالب کلی فرمهای شرطی به صورت زیر است:

(COND (P1 E1) (P2 E2) ... (PN EN))

مفسر دوتاییهای شرطی را از چپ به راست بررسی می‌کند. اگر P1 درست باشد، مقدار E1 را به عنوان نتیجه فرم شرطی باز می‌گرداند. در غیر این صورت، دوتایی بعد را بررسی می‌کند و در صورت لزوم بررسی دوتاییها را تا آخرین دوتایی ادامه می‌دهد.

گفتیم مفسر لیسپ چیزی جز یک تابع نیست که بر روی یک فرم (برنامه لیسپ) عمل می‌کند. این تابع EVALQUOTE نام دارد. تابع EVALQUOTE را می‌توان به زبان لیسپ نوشت. خواننده علاقمند به کسب اطلاعات بیشتر در مورد زبان لیسپ و مفسر آن، می‌تواند از مرجع (۲) این مقاله استفاده کند.

یادداشتها و مراجع

- * -Encyclopedia of Computer Science and Engineering, 1983, PP852-856
- * -LISP 1.5 J. MC CARTHY (MIT), MIT PRESS 1965
- * -"Programs, Machines, and Computa-

تا بهتها - که هر دو از انواع اتم هستند - و عبارات شرطی که بعداً "در مورد آنها صحبت خواهیم کرد.

از آنجایی که یک فرم می‌تواند نشانده‌ی فرم دیگری باشد، با استفاده از فرمهای تودرتو می‌توان برنامه‌های بزرگی ساخت. مفسر لیسپ چیزی جز یک تابع نیست. مفسر، یک فرم لیسپ را می‌خواند و سپس مقدار فرم را چاپ می‌کند.

ساده‌ترین فرم، یک متغیر است. یک متغیر عبارت است از هراتمی که یک مقدار به آن نسبت داده شده است. مقدار مذکور، اشاره‌گری به یک عبارت نمادین است. به عنوان مثال، مقدار یک متغیر X می‌تواند لیستی از سه عنصر به صورت (A B C) باشد. فرم اتمی دیگر لیسپ، مقدار دیرناست است. برخی از تا بهتای لیسپ عبارتند از: اعداد، T، و NIL. منظور از T، "درست" و منظور از NIL، "نادرست" است. همانطور که قبلاً گفتیم، ثابت NIL به نشانه انتهای لیست نیز به‌کار می‌رود.

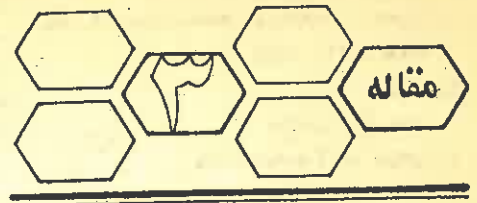
فرم QUOTE به صورت (QUOTE a) نوشته می‌شود. a یک عبارت نمادین است. مقدار (QUOTE a) است.

فرم تابعی در زبان لیسپ، فرم SUBR (مخفف subroutine) است که تابع گوناگونی را شامل می‌شود. CAR یک تابع SUBR است که نخستین عنصر یک لیست را بازمی‌گرداند. اگر (A B C) = x باشد، (CAR X) = A است. تابع CDR مشابه CAR عمل می‌کند، با این تفاوت که تمام لیست به جز عنصر اول آن را بازمی‌گرداند. اگر (A B C) = x باشد، (CDR X) = (B C) خواهد بود. برای جدا کردن هر قطعه دلخواه از یک عبارت نمادین، می‌توان از CARها و CDRهای تودرتو استفاده کرد. استفاده از اساسی CDR و CAR^۱ با بقیه تاریخی دارد. اینها دستورالعملهای زبان اسمبلی کامپیوتر آی بی ام ۷۰۴ هستند. کامپیوترنا میرده نخستین ماشینینی بود که زبان لیسپ روی آن پیاده شد.

تابع CONS (مخفف CONSTRUCT یا ساختار) نوع دیگری از توابع SUBR است. تابع CONS دو نشانه‌ی نمادین را، اگر A = Y و (CONS Y X) = (A A B C) خواهد بود.

تابع لاندا (LAMBDA) یکی دیگر از فرمهای لیسپ است. با استفاده از فرم لاندا می‌توان از یک عبارت، یک تابع ساخت. قالب کلی تابع لاندا به صورت زیر است:

(نشانوندها تعریف تابع لیست متغیرها (LAMBDA))



ترجمه و تلخیص: حمیدرضا رهبر

نگاهی برویژگیهای آدا

برای همیشه بلکه برای استفاده در مقاطع زمانی خاصی نیاز داریم و وزارت دفاع انگلستان اعلام نموده است که استفاده از آدا در پروژه‌های نظامی از اول ژوئیه ۱۹۸۷ الزامی است. ناتو نیز همین التزام را از ژانویه ۱۹۸۶ مورد تأکید قرار داده است.

● قابلیت مدور:

آدا یک زبان بین‌المللی تمام‌عیار است و در مقایسه با زبانهای بلادرنگ ۸ از قبیل CORAL و RTL/2 بسیار قابل قبولتر به نظر می‌رسد. همچنین درجهانی که همکاریها و مبادلات بین‌المللی از ضروریات آن است و هر کشوری زبان بلادرنگ مورد علاقه خود را دارد، آدا یک راه حل مشترک به شمار می‌رود.

● امکانپذیری:

با استفاده از آدا ساخت سیستمهایی که قبلاً به لحاظ اندازه و یا پیچیدگی غیرممکن تشخیص داده می‌شدند ممکن گشته است.

● پشتیبانی نرم‌افزاری سیستم:

امروزه شرکت‌های سازنده نرم‌افزار از زبانهای بلادرنگ قدیمی رویگردان شده‌اند اما مطمئناً آدا، حداقل به دلیل قدرت و اعتبار سازندگان، مدت زیادی توسط شرکت‌های نرم‌افزاری ارائه‌دهنده آن، پشتیبانی خواهد شد.

● طراحی برنامها:

تقریباً "می‌توان گفت که آدا اولین زبان برنامهنویسی است که عملاً" به عنوان یک زبان طراحی برنامها (PDL)^۹ می‌توان

بقیه در صفحه ۹

حرکت بعدی، تلاش در تعریف یک "محیط پشتیبانی برنامهنویسی آدا" (APSE)^۲ بود یعنی مجموعه‌ای به هم پیوسته از ابزار کمکی برای توسعه و پشتیبانی نرم‌افزار. این شاخه از علم کامپیوتر در مقایسه با طراحی زبانهای برنامهنویسی پیشرفت کمتری داشته است به همین دلیل استانداردهای طولانی APSE توسط وزارت دفاع هنوز راهی طولانی را در پیش دارد. این امر موجب شده است تا ارائه‌دهندگان سیستمهای نرم‌افزاری آدا، محیطهای پشتیبانی گوناگونی را با ابزار نرم‌افزاری متفاوت در اختیار استفاده‌کنندگان قرار دهند.

چرا آدا استفاده می‌شود؟

آدا، آخرین نظریه‌ها در فنون برنامهنویسی را به کار گرفته است. این زبان برای آنچه که جان بارنز^۳ آن را "برنامهنویسی حجیم" نامیده طراحی شده است یعنی برنامهنویسی و پشتیبانی نگهداری پروژه‌های متوسط تا فوق‌العاده بزرگ توسط متخصصان حرفه‌ای.

معمولاً یک یا چند مورد زیر، دلایل استفاده از آدا به شمار می‌روند:

● مهندسی نرم‌افزار:

برنامهای نوشته شده به زبان آدا، به راحتی "قابل تغییر" ۵، "قابل اطمینان" ۶ و "قابل درک" ۷ هستند و این سه قابلیت از ویژگیهای مهم برنامهای اصولی به حساب می‌آیند.

● نیازهای نظامی:

هر چند وزارت دفاع آمریکا به آدا نی

نرم‌افزار / پردازش داده‌ها / زبانهای برنامهنویسی / زبان آدا /

بیادداشت مترجم

زبان برنامهنویسی ADA در اصل برای کاهش هزینه‌های پشتیبانی و نگهداری (miantenance) نرم‌افزارهای وزارت دفاع آمریکا به وجود آمده است. "محیط پشتیبانی برنامهنویسی آدا" (APSE) نیز که متشکل از تعداد زیادی ابزار نرم‌افزاری جهت تسهیل برنامهنویسی به کمک آدا است در حال توسعه و آماده شدن می‌باشد.

در این مطلب کوتاه که برگزیده‌ای از مقاله مرجع است سعی شده تا مهمترین ویژگیهای آدا که باعث گرایش روزافزون شرکت‌های نرم‌افزاری و مراکز کامپیوتری به سوی این زبان شده‌اند، بر شمرده شوند.



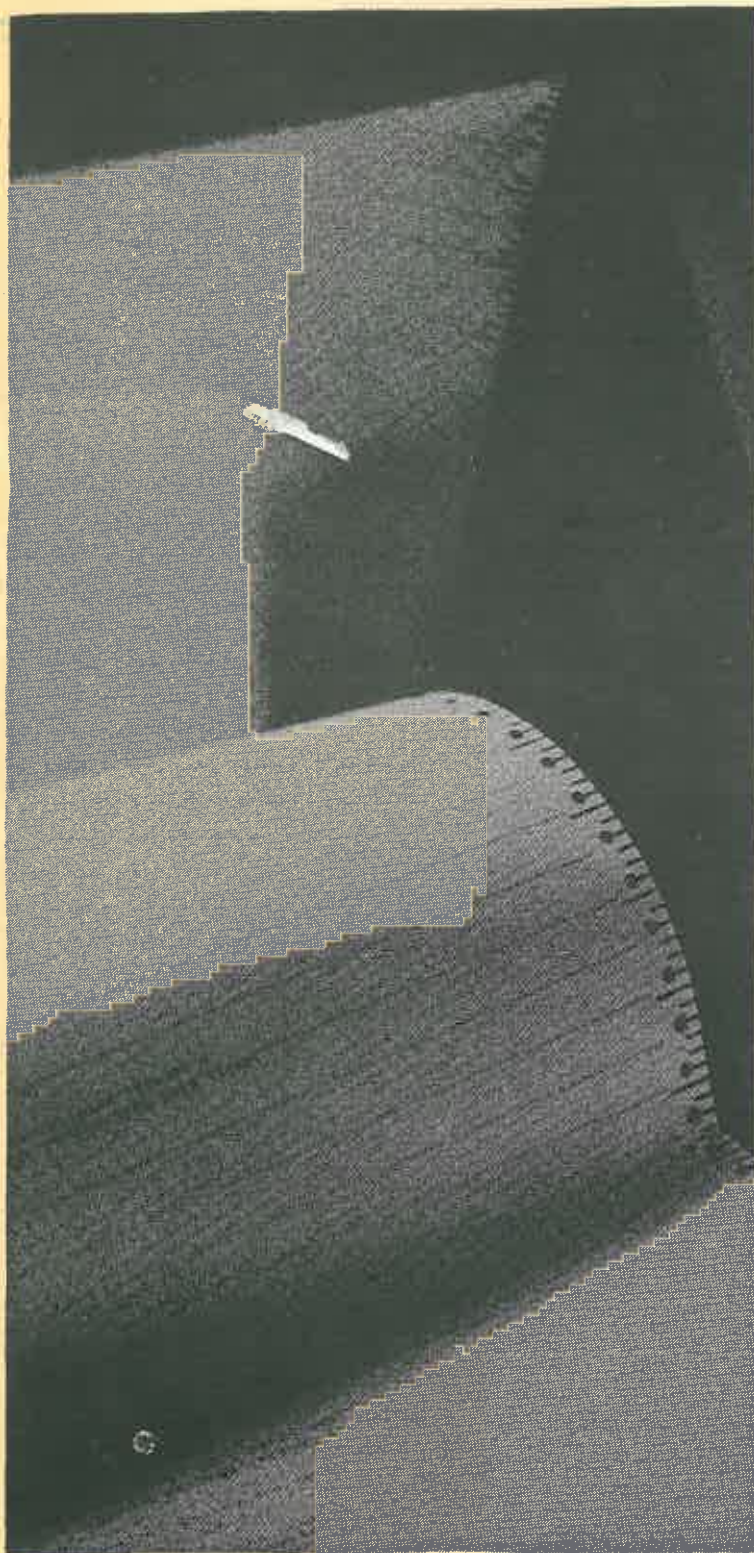
در سال ۱۹۷۳، وزارت دفاع آمریکا، برای مقابله با هزینه‌های هنگفت روبه‌افزایش نرم‌افزار و حرکتها و اقداماتی را آغاز نمود. کم‌کم یکی از این اقدامات به مسابقای بین‌المللی برای تولید یک زبان برنامهنویسی منجر شد به طوری که بتواند عمده‌ترین هزینه نرم‌افزاری وزارت دفاع یعنی هزینه نگهداری را کاهش دهد. در نهایت زبان جدیدی که ساخته یک گروه از متخصصان اروپایی بود برنده شناخته شد. این زبان به احترام خانم آدا اولیس، دستیاری از زبان بیج و اولین کسی که به طور آگاهانه برنامهنویسی را مطرح ساخت، "آدا" نامگذاری گردید.



نقطه شروع

مبتدیان

نوشته و اقتباس: مبتدیان با سیم



آشنائی با مفاهیم اساسی زبانهای برنامه سازی

نرم افزار / زبانهای برنامه نویسی / ساختها
وروشهای برنامه نویسی / برنامه نویسی
ساخته / مترجمها / الگوریتم

مقدمه

شما برای برقراری ارتباط با دوستتان چه میکنید؟ مسلماً "با استفاده از یک زبان مشترک با او گفتگو یا مکاتبه میکنید."

حتماً "میدانید که تفاوت اساسی کامپیوتر با ماشین حساب در این است که برخلاف ماشین حساب، میتواند روش انجام محاسبات و عملیات مورد نظر را برای کامپیوتر تشریح کرد. در این رابطه آیا تاکنون در مورد چگونگی برقراری ارتباط بین انسان و کامپیوتر اندیشیده اید؟ این عمل، با استفاده از زبانهای خاصی میسر میگردد که به زبانهای برنامه سازی مشهورند.

در این مقاله کوشش شده است مفاهیم اساسی زبانهای برنامه سازی، به زبان ساده برای مبتدیان تشریح گردد.

مفهوم الگوریتم

گفتیم روش انجام عملیات مورد نظر را با ایده نحوی برای کامپیوتر تشریح نمود. به روش انجام هر عمل، "مطالعه" "الگوریتم" آن عمل گفته میشود. به طور خلاصه میتوان الگوریتم را به صورت زیر تعریف کرد:

برای یک مسأله و یک دستگاه خاص، یک الگوریتم عبارت است از تشریح راه حل دقیق و مرحله ای مسأله به زبانی که برای آن



به روش انجام هر عمل، اصطلاحاً "الگوریتم" آن عمل گفته می‌شود.

```

10 READ X
20 IF X < 0 THEN 80
30 LET Y=1
40 LET Z=Y↑2
50 IF ABS (X-Z) < 0.00005 THEN 100
60 Y= ((X/Y)+Y)/2
70 GO TO 40
80 PRINT "THERE IS NO SOLUTION
FOR", X, "."
90 GO TO 10
100 PRINT "THE SQUARE ROOT OF", X,
"IS", Y, "."
105 GO TO 10
110 DATA 3,107,1,0,-4
120 END

```

شکل ۱: الگوریتم محاسبه جذرها را به زبان بیسیک (به نقل از مرجع دوم)

```

THE SQUARE ROOT OF 3 IS 1.73205
THE SQUARE ROOT OF 107 IS 10.3441
THE SQUARE ROOT OF 1 IS 1.
THE SQUARE ROOT OF 0 IS 3.90625E-3.
THERE IS NO SOLUTION FOR -4.

```

شکل ۲: نتیجه اجرای الگوریتم شکل ۱

زبانهای برنامه‌سازی

گفتیم که برای اجرای یک الگوریتم بر روی یک کامپیوتر، آن الگوریتم باید به زبانی که برای کامپیوتر قابل فهم باشد، نوشته شود. به چنین زبانهایی "زبانهای مصنوعی" یا "زبانهای برنامه‌سازی" گفته می‌شود. زبانهای مصنوعی از بسیاری جهات شبیه به زبانهای طبیعی هستند. این زبانها همانند زبانهای طبیعی القای خاصی دارند. با قراردادن کلمات زبانهای مصنوعی در کنار یکدیگر، بر طبق قواعد خاص، مانند دستور زبانهای طبیعی - می‌توان "جمله" یا "حکم" ساخت. احکام زبانهای مصنوعی درست مانند جملات زبانهای طبیعی دارای انواع مختلف هستند. انواع احکام متداول در زبانهای مصنوعی عبارتند از:

۱- حکم محاسبه‌ای و انتسابی: عملیات محاسبه‌ای و نسبت دادن مقادیر به

این روش وجود ندارد.

در روش زیر، سعی کرده‌ایم شکلات راه حل اول را تا حدودی بر طرف سازیم:

۱- عدد ۱ را به عنوان مقدار متغیر Y انتخاب کنید.

۲- مقدار Y را محاسبه کنید.

۳- اگر $|Y^2 - X| < 5 \times 10^{-5}$ باشد، Y جواب مسأله است، به اجرای الگوریتم پایان دهید. در غیر این صورت به مرحله بعد بروید.

۴- عدد $(X/Y + Y)/2$ را به عنوان مقدار جدید متغیر Y انتخاب کنید و به مرحله دوم بروید.

این راه حل، از راه حل قبلی بهتر است. اگر X یک عدد مثبت باشد، این راه حل مقدار آن را محاسبه می‌کند. اما اگر X منفی باشد، اجرای آن تا ابتدا دامنه‌ی بدوین با خاصیت اول الگوریتم مطابقت ندارد. برای تبدیل راه حل فوق به یک الگوریتم قابل اجرا، دو کار باید انجام شود:

۱- افزودن مرحله ۵: اگر $X < 0$ باشد، مسأله جواب ندارد. با ارسال پیغام مناسبی به اجرای الگوریتم پایان دهید.

۲- بدین ترتیبی راه حل فوق به زبانی که برای دستگاه مورد نظر، قابل فهم باشد (رجوع شود به تعریف اولیه‌ای که از الگوریتم اراکه دادیم). زبانی که الگوریتم محاسبه جذرها استفاده از آن نوشته شده است، برای فارسی زبانان مناسب و قابل فهم است.

اما برای اجرای الگوریتم بر روی یک کامپیوتر، باید از زبانی که برای کامپیوتر قابل فهم باشد استفاده نمود. به عنوان مثال الگوریتم شکل ۱ که به زبان بیسیک نوشته شده است، برای کامپیوترهایی که این زبان را "درک" می‌کنند، مناسب است. هدف از اجرای این الگوریتم، محاسبه جذر اعداد ۴- و ۰ و ۱ و ۱۰۷ و ۳ می‌باشد.

توجه کنید که دستورالعمل ۱۱۰، مجموعه داده‌های مورد استفاده الگوریتم را مشخص می‌سازد و جزء الگوریتم نیست. نتیجه اعمال الگوریتم فوق بر روی مجموعه داده‌های مذکور به صورت شکل ۲ است.

"زبانهای مصنوعی" یا

زبانهای برنامه‌سازی

حدواسط زبانهای طبیعی

وزبانهای ماشین هستند که

برای پیاده‌سازی

الگوریتمها به کار می‌روند.

دستگاه قابل فهم باشد. الگوریتم، یک "دنباله" یا "سلسله" است.

هر الگوریتم باید شرایط زیر را برآورده سازد:

۱- دنباله (سلسله) دستورالعملهای الگوریتم باید متناهی باشد.

۲- اولین دستورالعمل دنباله، باید یک دستورالعمل یگانه باشد. به عبارت دیگر، نقطه شروع سلسله دستورالعملها مشخص و منحصر به فرد باشد.

۳- دستورالعمل بعدی هر دستورالعمل دنباله، باید یگانه باشد یعنی پس از هر دستورالعمل، نقطه و پله بعدی مشخص و منحصر به فرد باشد.

۴- دنباله دستورالعملها باید به حل مسأله و یا به جمله‌ای که غیر قابل حل بودن مسأله را بیان می‌کند، منتهی گردد.

ذکر یک مثال به روشن شدن مفهوم الگوریتم کمک می‌کند. فرض کنید می‌خواهیم الگوریتمی برای محاسبه جذر یک عدد، با چهار رقم اعشار، بنویسیم. یک روش برای حل این مسأله ممکن است به صورت زیر باشد:

۱- یک عدد Y را انتخاب کنید و Y را محاسبه نمایید.

۲- اگر $|Y^2 - X| < 5 \times 10^{-5}$ باشد، عدد Y پاسخ مسأله است. در غیر این صورت به مرحله اول بازگردید.

دنباله دستورالعملهای فوق نمی‌تواند یک الگوریتم باشد، زیرا چگونگی انتخاب مقدار اولیه و مقدار دیربندی Y در آن بیان نشده است. این امر با تعریف الگوریتم که تشریح دقیق راه حل مسأله است، مغایرت دارد. علاوه بر آن، حتی در صورت وجود جواب، هیچ تضمینی برای یافتن آن با استفاده از

به طور کلی، به برنامهای که تنها از چهار ساختار فوق استفاده می کنند، برنامه های ساختار یافته گفته می شود و این در مقابل برنامه های بی ساختار است که در آنها پرش های شرطی و بدون قید و شرط به طور پراکنده وجود دارد و در حقیقت به جای استفاده از ساختار را شعاب شرطی و ساختارهای حلقه ای، دارای پرش های آزاد به صورت احکام مستقل و پراکنده هستند. برای مثال برنامه محاسبه جذر (شکل ۱) را نمی توان ساختار یافته دانست. چرا؟

می توان اثبات کرد که هر برنامه مدللخواه، معادل یک برنامه تکراری شرطی است. به عبارت دیگر هر برنامه را می توان به صورت ساختار یافته نوشت. مهمترین محاسن برنامه های ساختار یافته عبارتند از: سهولت اثبات درستی برنامه ها و آسان شدن فهم آنها.

مترجم انواع آن

برای گفتگوی کسی که زبان شما آشنا نیست، چه می کنید؟ بدیهی است که از مترجمی که زبان شما و مخاطب شما آشناساز استفاده می کنید. برنامه محاسبه جذر که به زبان بیسیک نوشته شده بود، مستقیماً توسط کامپیوتر قابل فهم نیست. بنا بر این وجود عاملی که آن را به زبان قابل فهم توسط ماشین تبدیل کند، ضروری است. این ماشین عامل، برنامه مترجم و عملی که انجام می دهد ترجمه نامیده می شود. زبان قابل فهم توسط ماشین نیز اصطلاحاً "زبان ماشین" خوانده می شود. به عبارت دیگر، با استفاده از زبان ماشین، می توان دستورالعملها و داده ها را به شکلی که بلافاصله توسط سخت افزار کامپیوتر قابل تفسیر و اجرا باشد، نوشت.

آیا می توانید بگویید چرا به جای نوشتن برنامه جذر به زبان بیسیک آن را به زبان ماشین ننوشتیم؟ پاسخ این است که نوشتن برنامه به زبان ماشین بسیار دشوار و وقتگیر است. به عنوان مثال، دستورالعمل "کپی کردن محتوای ثبات A در ثبات X" در زبان ماشین کامپیوتر HP 21 MX به صورت زیر است:

```
1000001111101001
```

حتماً "تصدیق می کنید که نوشتن برنامه به چنین زبانی که دستورالعملهای آن رمزهای طولانی متشکل از ارقام ۰ و ۱ هستند، کار بسیار طاقت فرسایی است. خوب است بدانید که نخستین برنامه سازان جهان، برنامه های خود را به چنین زبانهایی می نوشتند. پس از

بقیه در صفحه ۲۰

به کار می روند. منظور از عملیات ورودی و خروجی، دادن اطلاعات لازم به کامپیوتر (برای اعمال الگوریتم بر روی آنها) و اخذ نتایج اجرای الگوریتم از کامپیوتر است. در برنامه محاسبه جذر، حکم شماره ۱۰ یک حکم ورودی است و احکام شماره ۸۰ و ۱۰۰ از نوع احکام خروجی هستند.

ساختارها و برنامه سازی ساختار یافته

از ترکیب احکامی که بر شمر دیسک، ساختارهای گوناگون به وجود می آیند. به طور کلی، هر الگوریتم یا برنامه را می توان به ساختارهای زیر نوشت:

۱- ساختار توالی یا الحاق خطی: دنباله ای از احکام که به ترتیب یکی پس از دیگری اجرا می شوند، یک ساختار توالی یا الحاق خطی را تشکیل می دهند. به عنوان مثال احکام شماره ۳۰ و ۴۰ در برنامه محاسبه جذر، یک ساختار توالی را می سازند.

۲- ساختار انتخاب یا اشعاب شرطی: هر حکم شرطی همراه با دنباله احکام وابسته به آن (دنباله احکامی که در صورت برقراری یا عدم برقراری شرط باید اجرا گردند) یک ساختار انتخاب یا اشعاب شرطی را تشکیل می دهد و این در صورتی است که احکام وابسته به شرط، مشخص و یا بیان آنها تعیین شده باشد.

آیا در برنامه محاسبه جذر چنین ساختاری وجود دارد؟

۳- ساختار حلقه های شمارشی: دنباله ای از احکام که در یک حلقه تکرار شمارشی قرار دارند، چنین ساختاری را به وجود می آورند.

۴- ساختار حلقه های تکراری شرطی: دنباله ای از احکام واقع در یک حلقه تکرار شرطی، یک ساختار حلقه های تکراری شرطی را تشکیل می دهند.

هر برنامه را با توجه به نوع ساختارهای به کار رفته در آن، می توان به یکی از انواع زیر تقسیم کرد:

۱- برنامه های شرطی: برنامه هایی که تنها ساختارهای الحاق خطی و اشعاب شرطی در آنها به کار رفته است.

۲- برنامه های شمارشی: برنامه هایی که تنها شامل ساختارهای الحاق خطی، اشعاب شرطی، و حلقه شمارشی هستند.

۳- برنامه های تکراری شرطی: برنامه هایی که مشتمل بر چهار ساختار نامبرده باشد، برنامه های تکراری شرطی خوانده می شوند.

متغیرها توسط این احکام انجام می شود. احکام شماره ۳۰، ۴۰ و ۵۰ در برنامه محاسبه جذر نمونه ای از این گونه احکام هستند. ۲- حکم شرطی: این گونه احکام بر حسب برقرار بودن یا نبودن شرط خاصی، موجب می گردند که اجرای برنامه از یکی از دو نقطه مختلف برنامه ادامه یابد. قالب این احکام در اکثر زبانهای برنامه سازی به صورت زیر است:

دنباله احکام ۱ THEN شرط IF
دنباله احکام ۲ ELSE

حکم شماره ۵۰ در برنامه محاسبه جذر، از انواع احکام شرطی است.

۳- حکم تکرار: با توجه به الگوریتم محاسبه جذر در میابید که دستورالعملهای ۳۰، ۴۰ و ۵۰ یک حلقه تکرار را تشکیل می دهند. به عبارت دیگر این سه دستورالعمل ممکن است بارها اجرا شوند. آیا می توانید در برنامه محاسبه جذر، یک حلقه تکرار پیدا کنید؟

حکم تکرار، احکامی را که باید چندین بار تکرار شوند، در حلقه ای موسوم به "حلقه تکرار" قرار می دهد. حلقه های تکرار - و در نتیجه احکام وابسته به آنها - را می توان به دو نوع شرطی و شمارشی تقسیم کرد.

حلقه های تکرار شمارشی^۶: در حلقه های تکرار شمارشی، دنباله ای از احکام که در داخل حلقه قرار دارند، به تعداد دفعات معین تکرار می گردند. قالب احکام تکراری شمارشی در زبان برنامه سازی "پیل وان" ^۷ به صورت زیر است:

مقدار انتهایی T0 مقدار اولیه = شمارنده DO
احکامی که باید تکرار شوند.

END;
اگر مقدار اولیه و مقدار انتهایی برابر باشد، احکام داخل حلقه، به تکرار خواهند شد.

حلقه های تکرار شرطی^۸: در حلقه های تکرار شرطی، احکام داخل حلقه تا وقتی که شرط خاصی برقرار است، تکرار می شوند. قالب احکام تکراری شرطی در زبان پیل وان به صورت زیر است:

DO WHILE شرط
احکامی که باید تکرار شوند {

اگر شرط تکرار حلقه، $I > 0$ باشد، تا زمانی که متغیر I از صفر بزرگتر است، اجرای احکام داخل حلقه تکرار می گردد. ۴- حکم ورودی و خروجی: این دسته از احکام برای تشریح عملیات ورودی و خروجی



در سال ۱۹۵۹ تشکیل این فدراسیون و

قوانین آن مورد تصویب یونسکو قرار گرفت

آشنایی با سازمانها و مراکز کامپیوتری جهان

فدراسیون بین المللی پردازش اطلاعات (IFIP)

Country	Society
Algeria	Commissariat National a l'Informatique
Argentina	Sociedad Argentina de Informática
Australia	Australian Computer Society
Austria	Austrian Computer Society
Belgium	FAIB-FBVI
Brazil	Sociedade dos Usuários de Computadores e Equipamentos Subsidiários (SUCESU)
Bulgaria	Bulgarian Academy of Sciences
Canada	Canadian Information Processing Society
China, Peoples Rep. of	Chinese Institute of Electronics
Cuba	Academia de Ciencias de Cuba
Czechoslovakia	Czechoslovak National Committee
Denmark	Danish Federation for Information Processing
Egypt	Egyptian Computer Society
Finland	Finnish Association for Data Processing
France	Association Française pour la Cybernétique Economique et Technique (AFCET)
German Democratic Republic	Academy of Sciences of the German Democratic Republic
Hungary, Federal Republic of	Deutsche Arbeitsgemeinschaft für Rechenanlagen (DARA)
India	John von Neumann Society
Iraq	Computer Society of India
Ireland	Planning Board/National Computer Centre
Israel	Irish Computer Society
Italy	Information Processing Association of Israel (IPA)
Japan	Associazione Italiana per il Calcolo Automatico (AICA)
Korea, Rep. of	Information Processing Society of Japan
Morocco	Korea Information Science Society
	Association Marocaine pour le Développement de l'Electronique et de l'Automatique (AMADEIA)
Netherlands	Nederlands Rekenmachine Genootschap
New Zealand	The New Zealand Computer Society
Nigeria	Computer Association of Nigeria
Norway	Norwegian Computer Society
Poland	Polish Academy of Sciences
Portugal	Associação Portuguesa de Informática
South Africa	Computer Society of South Africa
Spain	Federacion Espanola de Sociedades de Informatica
Sweden	Swedish Society for Information Processing
Switzerland	Federation Suisse d'Informatique
Tunisia	Centre National de l'Informatique
United Kingdom	British Computer Society
U.S.A.	American Federation of Information Processing Societies (AFIPS)
USSR	The Computing Centre of the USSR Academy of Sciences
Yugoslavia	Yugoslav Committee for Electronics and Automation (ETAN)

جدول شماره ۱

این کمیته از کشورهای آمریکا، فرانسه، سوئد، شوروی، اسپانیا، آلمان غربی، هلند، انگلستان و ژاپن بودند. در اولین کنفرانس بین المللی پردازش اطلاعات که در ماه ژوئن سال ۱۹۵۸ در پاریس برگزار گردید، نمایندگان ۱۸ انجمن کامپیوتر از سراسر دنیا طی جلسات مشترک رایج به ساختار این فدراسیون به تبادل نظر پرداخته و قوانین اولیه آن را

تاریخچه پیدایش: در طی سالهای ۱۹۵۰ در جلسه های خصوصی که برای بحث، تبادل نظریات و مباحثه ریزی در زمینه پردازش اطلاعات تشکیل میشد نیاز به تشکیل یک مجمع بین المللی برای برنامهریزی و توسعه علوم اطلاعات احساس میگردید. جهت برآوردن این نیاز کمیته ای تحت سرپرستی آقای ایواک آشر باخ (از آمریکا) تشکیل گردید تا قوانین اولیه چنین مجمعی را تهیه کرده و زمینه را برای فعالیتهای آن آماده سازد. اعضای

ترجمه: دکتر محسن رستمی

۱/ انفورماتیک در کشورها / سازمانهای کامپیوتری / IFIP

فدراسیون بین المللی پردازش اطلاعات یک مؤسسه بین المللی متشکل از انجمنهای حرفه ای - علمی جهان است که در زمینه پردازش اطلاعات فعالیت دارند. بر طبق اساسنامه این فدراسیون از هر کشور یک انجمن میتواند در این مؤسسه عضویت داشته باشد. هر انجمن عضو باید نماینده کلیه فعالیتهای کشورش در زمینه پردازش اطلاعات باشد.

این فدراسیون تا ابتدای سال ۱۹۸۰ دارای ۳۹ عضو بود. اساسی این ۳۹ عضو در جدول شماره ۱ نشان داده شده اند.

اهداف فدراسیون بین المللی پردازش اطلاعات به شرح زیر است:

- ۱- ارائه و توسعه علم و تکنولوژی اطلاعات در سطح بین المللی
- ۲- تشویق و پیشرفت همکاریهای بین المللی در زمینه پردازش اطلاعات
- ۳- تشویق و ترغیب تحقیقات در زمینه پردازش اطلاعات و کاربردهای آن در علوم و فعالیتهای بشری
- ۴- گسترش مبادله اطلاعات در زمینه پردازش اطلاعات
- ۵- تشویق تحصیلات در زمینه پردازش اطلاعات

این فدراسیون علاوه بر فعالیت مهمی که در پیشبرد علم پردازش اطلاعات دارد همچنین نقش مؤثری در گسترش و تبادل اطلاعات در زمینه وسایل این علم یعنی کامپیوترهای رقمی و کاربردهای آنها نیز داشته است.

از هر کشور جهان فقط یک انجمن می‌تواند در این فدراسیون عضویت داشته باشد و باید کلیه فعالیت‌های انفورما تیکی کشورش را نمایندگی کند.

است و بر جنبه‌های کاربردی و تجاری پردازش اطلاعات تأکید دارد، تا به حال در سطح بین المللی فعالیت‌ها و تجربیات مؤسسه در زمینه‌های تحقیقات، آموزش و تبادل اطلاعات داشته است.

IMIA عضو وابسته دیگر این فدراسیون که در کمیته فنی شماره ۴ این سازمان شرکت دارد روی زمینه کاربرد کامپیوتر در پردازش اطلاعات پزشکی تأکید دارد. IMIA تا به حال چند کنفرانس بین المللی^{۱۳} در این زمینه تشکیل داده است که سومین آنها در سال ۱۹۸۰ در توکیو ژاپن برگزار گردید.

فعالیت‌های فدراسیون در مجامع بین المللی

این فدراسیون تا به حال با سازمان یونسکو رابطه‌ای داشته و با این سازمان در رده B همکاری دارد یعنی می‌تواند در زمینه‌های نظرات و مشورت‌های تخصصی ارائه دهد.

این فدراسیون همچنین از سال ۱۹۷۲ با سازمان جهانی بهداشت رابطه رسمی داشته و با بیشتر اعضای دیگر سازمان ملل متحد نیز رابطه غیر رسمی و همکاری دارد. فدراسیون همچنین به صورت مشاوره کمیته بین المللی تلفن و تلگراف (CCITT) همکاری دارد. در سال ۱۹۷۰ این فدراسیون با چهار فدراسیون بین المللی دیگر به نام‌های IMACS^{۱۴}، IFAC^{۱۵}، IFORS^{۱۶} و IMEKO یک کمیته مشترک پنجگانه برای برنام‌های ریزبینی مشترک و تبادل اطلاعات در گردهما شیهی سالانه تشکیل داده است.

کنگره‌های فدراسیون:

برجسته‌ترین فعالیت این فدراسیون برگزاری یک کنگره بین المللی است که هر سه سال یکبار برگزار می‌گردد. در این کنگره‌ها متخصصین علم پردازش اطلاعات مدیران و سایر افراد علاقه‌مند به این علم فرصت شنیدن، فراگیری و تبادل اطلاعات با دیگر همکاران خود از سایر کشورها را خواهند داشت. اولین کنگره این فدراسیون در سال ۱۹۶۲ در شهر مونیخ در آلمان غربی برگزار گردید. در اکتبر سال ۱۹۸۰ این کنگره به طور همزمان در شهرهای توکیو ژاپن و ملبورن استرالیا برگزار شد.

بقیه در صفحه ۷

کمیته‌های فنی فدراسیون:

برای بسط و پیشبرد علم پردازش اطلاعات این فدراسیون اقدام به تشکیل کمیته‌های فنی^۷ مختلف در این زمینه کرده است. هر انجمن کامپیوتر عضو این فدراسیون می‌تواند در هر کمیته فنی یک عضو داشته باشد. هر کمیته فنی از چند گروه کاری تخصصی^۸ که تحت نظر آن کمیته کار می‌کنند تشکیل شده است. اعضای تشکیل دهنده هر گروه کاری با توجه به تخصص خاص آنها انتخاب شده و ملیت آنها مورد نظر قرار نمی‌گیرد.

بعضی از کمیته‌های فنی و گروه‌های کاری وابسته به آنها به شرح زیر می‌باشند:

- کمیته فنی شماره ۲- زبانهای برنامه نویسی
- گروه کاری "زبان الگول"
- گروه کاری "توصیف فرمال مقایسه برنامه نویسی"
- گروه کاری "سبک‌شناسی برنامه نویسی"
- گروه کاری "پایگاه‌های داده‌ها"
- گروه کاری "فصل‌های سیستم عامل"

⋮

کمیته فنی شماره ۳- آموزش

- گروه کاری "آموزش انفورما تیک در سطح دبیرستان"
- گروه کاری "پروژه‌های طرح درس‌های پیشرفته در زمینه پردازش اطلاعات"
- گروه کاری "آموزش حرفه‌ای کامپیوتر"

⋮

کمیته فنی شماره ۴- ارتباط داده‌ها^{۱۰}

- گروه کاری "ارتباط بین انسان و کامپیوتر"
- گروه کاری "شبکه‌های محلی کامپیوتر^{۱۱}"
- گروه کاری "سیستم‌های خبری کامپیوتری بین المللی^{۱۲}"

⋮

کمیته فنی شماره ۸- سیستم‌های اطلاعاتی

- گروه کاری "طراحی و برآورد سیستم‌های اطلاعاتی"
- گروه کاری "ارتباط بین سازمان و سیستم اطلاعاتی"

⋮

۱۱ IAG که از اعضای وابسته فدراسیون

مورد مرور و بررسی قرار دادند. چند ماه پس از برگزاری این کنفرانس این قوانین توسط ۱۳ انجمن کامپیوتر مورد تصویب قرار گرفت (حدقل ۷ رأی مثبت برای تصویب نهایی این قوانین لازم بود).

در دومین کنفرانس بین المللی پردازش اطلاعات که در سال ۱۹۵۹ توسط یونسکو برگزار شد، تشکیل این فدراسیون و قوانین آن مورد تصویب یونسکو نیز قرار گرفت.

تشکیلات سازمانی و اداری این فدراسیون در اول ماه ژانویه سال ۱۹۶۰ آغاز به کار کرد. بعضی از رؤسای این فدراسیون از بدو پیدایش تا کنون به شرح زیر می‌باشند:

- ایزاک آشر باخ ۱۹۶۵-۱۹۶۰ از آمریکا
- دورودنی سین ۱۹۷۱-۱۹۶۸ از شوروی
- هانتس زانک ۱۹۷۲-۱۹۷۱ از اتریش
- پیر بوبی لیز ۱۹۷۷- از سوئیس

ساختار فدراسیون:

بالاترین مرجع تصمیم گیری در این فدراسیون مجمع عمومی است که سالی یکبار تشکیل جلسه می‌دهد. اعضای این مجمع به شرح زیرند:

- یک نماینده از هر انجمن عضو (اعضای اصلی)
- رؤسای دو عضو وابسته به نام‌های IAG^۳ و IMIA^۴
- رؤسای دو عضو وابسته دیگر به نام‌های IAPR^۵ و IASC^۶
- سه عضو افتخاری دائمی (ایزاک آشر باخ، هانتس زانک و ریچارد ناگا)
- هیئت اجرایی این فدراسیون از اعضای زیر تشکیل شده است:
- یک رئیس
- سه نایب رئیس
- یک دبیر
- یک خزانه دار

اعضای هیئت اجرایی توسط مجمع عمومی انتخاب می‌شوند. کارهای روزمره کنفدراسیون توسط یک منشی که در سوئیس مستقر است صورت می‌پذیرد. گروهی شامل هیئت اجرایی و (تا) هشت نفر هیئت امناء سالی دوبار تشکیل جلسه داده و مواردی را که قبل از تشکیل مجمع عمومی بایدراجع به آنها تصمیم گیری شود مورد بررسی قرار می‌دهد.



گزارش

گزارش فعالیتهای یکساله انجمن انفورماتیک ایران

(مهر ۱۳۶۴ تا شهریور ۱۳۶۵)

انجمن از طریق ستون "پاسخ به سؤالات" در ماهنا مه بوده است.

ب- کمیته فعالیتهای علمی و فنی

اهم فعالیتهای این کمیته طی سال گذشته به قرار زیر بوده است:

- ۱- ترتیب دادن سخنرانی علمی به شرح زیر: گزارشی از نمايشگاه علمی سوکوبا - ژاپن (مهر ماه ۱۳۶۴)، کشف مفاهیم مجرد در ریاضی توسط کامپیوتر (بهمن ماه ۱۳۶۴)
- ۲- پاسخگویی به سؤالات علمی و فنی اعضای انجمن به طور خصوصی (تلفنی، حضوری و مکاتباتی) یا از طریق ستون "پاسخ به سؤالات" در ماهنا مه گزارش کامپیوتر (با کمک کمیته انتشارات و دیگر اعضای هیئت اجرایی انجمن).

۳- همکاری در برگزاری کنفرانس بین المللی یونسکو در زمینه پردازش زبانهای بومی که قرار است در بهار سال آینده برگزار گردد.

۴- دریافت تشریفات و کتب جدید برای کتابخانه انجمن و تهیه فهرست آنها جهت درج در ماهنا مه گزارش کامپیوتر. لازم به تذکر است که در سال گذشته نیز فعالیتهایی جهت یافتن محل مناسبی برای دایر نمودن کتابخانه انجمن صورت پذیرفت که متأسفانه تا کنون موفقیت آمیز نبوده است.

۵- همکاری در برگزاری سمپوزیوم نوروپسیکولوژی در اردیبهشت ماه ۱۳۶۵ (با کمک سایر افراد هیئت اجرایی).

ت- کمیته عضویت

کمیته عضویت انجمن نیز در سال گذشته از پرمشغله ترین کمیته های انجمن بود. لازم به تذکر است که از فروردین ماه ۱۳۶۵

جایزه خوارزمی سال ۱۳۶۴ را در دو گسریه آزاد و دانشجویی معرفی نمود. جوایز برندگان، همزمان با برگزاری مجمع عمومی سالانه انجمن طی مراسمی به آنان اهداء خواهد گردید.

۵- صافی ماهنا مه های سال ۱۳۶۴ در ۴۳۳ نسخه. بهای این دوره جلد شده، برای اعضای انجمن ۱۵۰۰ ریال و برای سایرین ۱۵۰۰ ریال تعیین گردیده است.

۶- اعمال تغییرات و تجدید چاپ بروشور فعالیتهای و فرم عضویت در انجمن.

۷- تلاش در جهت اخذ سهمیه دولتی کاغذ.

۸- اعمال تغییراتی در شکل و چهره زجوب ظاهری نشریه از جمله جلد و زکسردن آن، تغییر صفحه بندی، انتخاب جملات برجسته و کلمات کلیدی مقالات.

۹- ایجاد آرشیه های طرح، مقاله و خبر.

۱۰- افزایش تیراژ نشریه جهت تأمین تعداد بیشتری دوره جلد شده.

۱۱- ایجاد دستونه های جدیدی در نشریه از جمله "نقطه شروع مبتدی" و "آشنایی با سازمانها و مراکز کامپیوتری".

ب- کمیته آموزش

کمیته آموزش انجمن در سال گذشته علیرغم پربار مهربانی برای برگزاری دوره های آموزشی مختلف، به دلیل امکانات مالی محدود انجمن و در اختیار نبودن محل مناسبی برای برگزاری این دوره ها، در این زمینه فعالیت شایان توجهی نداشت. تنها دوره آموزشی ترتیب داده شده توسط این کمیته در شهریور ماه ۱۳۶۵، تحت عنوان "تجزیه و تحلیل سیستمها" در دانشگاه صنعتی امیرکبیر تهران به مدت ۱۴ روز برگزار گردید.

از دیگر اقدامات کمیته آموزش انجمن ظرف سال گذشته، تهیه مقالات مناسب با جنبه آموزشی برای ماهنا مه گزارش کامپیوتر و نیز پاسخگویی به سؤالات علمی و فنی اعضای

آنچه در زیر به نظر اعضای گرامی انجمن میرسد، خلاصه ای از فعالیتهای گوناگونی است که توسط کمیته های مختلف انجمن ظرف سال گذشته صورت گرفته است. امید است با همکاری هر چه بیشتر و پربارتر اعضا و تلاش و کوشش شریکانشرا فرا دهیست اجرائی، کارنامه فعالیتهای سالانه انجمن در آینده، پربارتر گردد.

الف- کمیته انتشارات

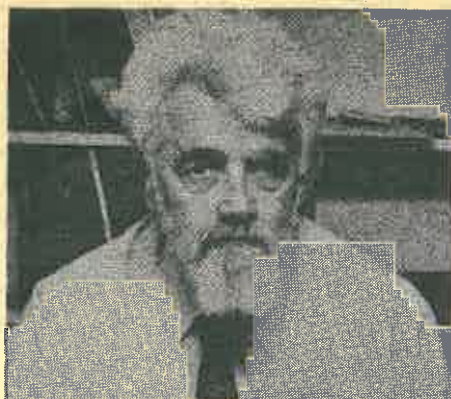
کمیته انتشارات در سال گذشته نیز بر کار رترین کمیته انجمن بود و با زده مشخص آن که انتشار ماهنا مه گزارش کامپیوتر می باشد، در معرض مطالعه و قضاوت اعضای گرامی انجمن قرار گرفته است. اهم فعالیتهای کمیته انتشارات در یکسال گذشته به شرح زیر بوده است:

۱- انتشار هشماره ها ماهنا مه های گزارش کامپیوتر (مهر ۱۳۶۴ - شماره پیاپی ۶۹، تا تیر و مرداد ۶۵ - شماره پیاپی ۷۸). لازم به تذکر است که هشماره اول این دوره در زمان سرپرستی مسئول قبلی کمیته و بقیه در زمان سرپرستی مسئول فعلی کمیته انتشارات به چاپ رسیده است و هشماره شهریور و مهر ۶۵ نیز که بطوریکه چاپ شده اند در محاسبه فوق منظور نگشته اند. در این هشماره، جمعا ۵۵ مقاله علمی و فنی (بدون احتساب دنباله مقالات چند قسمتی) همراه با مطالب متنوع و گوناگون خبری و تخصصی در ۴۸ صفحه به چاپ رسیده است.

۲- جهت دادن به مطالب و موضوعات چاپ شده در ماهنا مه.

۳- ایجاد دهیست تحریریه و گروه ویراستاران.

۴- انتخاب برندگان جایزه خوارزمی سال ۱۳۶۴. در اوائل سال جاری، هیئت داوران منتخب هیئت اجرایی انجمن، به بررسی مقالات چاپ شده پرداخت و برندگان



آشنایی با
پیشتان
علم کامپیوتر

قسمت : ۲۵

جان مک کارتی : خالق زبان لیسپ و مبتکر اصطلاح هوش مصنوعی

مک کارتی برنده جایزه معروف تورینگ در سال ۱۹۷۱ است و در سال ۱۹۸۵ از سوی IEEE به عنوان یکی از پیشتا زان علم کامپیوتر برگزیده شده است .

که در مقاله معروف سال ۱۹۵۸ وی با عنوان "برنامه های دارای عقل سلیم" ^{۱۱} برای اولین بار ارائه گردید، واقعیات جهان مورد بررسی و نتایج مورد نظر را به صورت جملاتی در منطق ریاضی درمی آورد و از راه جستجوی جهت دار، مراحل نیل به هدف را نتیجه می گیرد .

مک کارتی در دهه ۱۹۶۰ آنچه را که "حساب وضعیت" ^{۱۲} نامیده می شود با به گذاری کرد، این حساب در تمایش آثار عملیات مختلف بر روی جهان در دست بررسی مفید است و مبنای کار بر برنامه های هوشمندی را که برای انجام عملیات مختلف به منظور نیل به هدفهای مشخص طرح می ریزند، تشکیل می دهد. در دهه ۱۹۷۰ وی روش "محدود سازی" ^{۱۳} را در استدلال غیریکتاخت ^{۱۴} به عنوان مکملی برای استنتاج منطقی پی گیری کرد و شکل تکامل یافته آن را در سال ۱۹۸۰ در مجله "هوش مصنوعی" منتشر ساخت .

اگرچه ادعای روش اشتراک و قس ^{۱۵} عموماً به فرد دیگری نسبت داده می شود، اما دکتر مک کارتی نیز در شکل گیری این روش نقش موثری داشته است. اولین سیستم مشهور اشتراک وقت، که بر روی کامپیوتر آی بی ام ۷۰۹۴ پیاده شده بود، در پائیز سال ۱۹۶۳ در دانشگاه ام آی تی به کار افتاد، حال آنکه پیش از آن، یعنی در تابستان سال ۱۹۶۲، سیستم اشتراک وقتی که تحت رهبری دکتر

بقیه در صفحه ۱۹

مک کارتی از سال ۱۹۴۹ به مک انیمهائی که توسط آنها بتوان ماشینها را قادر به تفکر نمود علاقمند شد و از همان ابتدا به مشکل بودن این مسأله پی برد. اولین نوشته او در این زمینه در کتابی که با کلود شانون مسئولیت ویراستاری آن را بر عهده داشت ^{۱۶} به چاپ رسید .

در کالج دارتموث، توجه مک کارتی از ریاضیات کاربردی به علم کامپیوتر منحرف شد. در تابستان سال ۱۹۵۶ بود که مک کارتی در رابطه با زماندهی یک گره هم آئشی پژوهشی، اصطلاح "هوش مصنوعی" را ابداع کرد. در این گره هم آئی، افراد علاقمند به هوشمند کردن کامپیوتر توانستند برای اولین بار به همفکری و تبادل نظر بپردازند و با یکدیگر متحد شوند.

مک کارتی با استفاده از افکار آلن نیوول و هربرت سایمون ^{۱۷}، مبتکران زبان آی بی ال ^{۱۸}، و نیز با بهره گیری از تجربیات جان باکوس ^{۱۹} در طراحی زبان فرترن ^{۲۰}، شروع به کار بر روی یک زبان برنامه سازی برای محاسبه بر روی عبارتهای نمادین ^{۲۱} کرد. اولین نتیجه این پژوهش زبان "فرترن برای پردازش لیستها" ^{۲۲} بود، اما نتیجه اصلی که در سال ۱۹۵۸ ظاهر گردید، معرفی زبان لیسپ بود .

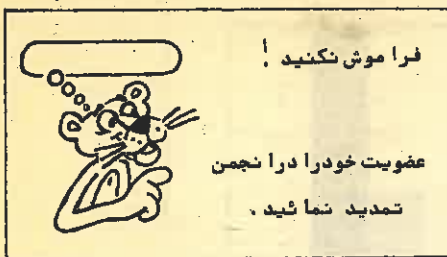
مهمترین آثار علمی دکتر مک کارتی به برنامه های کامپیوتری "استدلال گستر" مربوط می شوند. روش پیشنهادی مک کارتی

ترجمه : دکتر بهروز پیرها می
استاد میهمان دانشگاه واترلو

داشتمندان کامپیوتر / هوش مصنوعی /
زبانهای برنامه نویسی / زبان لیسپ /

جان مک کارتی در چهارم سپتامبر سال ۱۹۲۷ میلادی در شهر بوستون ایالات متحده آمریکا به دنیا آمد. در سال ۱۹۴۸ از انستیتوی تکنولوژی کالیفرنیا ^{۲۳} در رشته ریاضی فارغ التحصیل شد و در سال ۱۹۵۱ درجه دکتری ریاضی خود را از دانشگاه پرینستون گرفت. پس از دریافت درجه دکتری، دو سال به عنوان مدرس ریاضیات در دانشگاه پرینستون خدمت کرد و سپس با عنوان "استاد موقت" ^{۲۴} به استخدام دانشگاه استنفورد درآمد.

دکتر مک کارتی در سال ۱۹۵۵ به کالج دارتموث ^{۲۵} رفت و سپس در سال ۱۹۵۷ به انستیتوی تکنولوژی ماساچوست (ام آی تی) پیوست و در همان جا به مرتبه دانشیار ^{۲۶} ارتقاء یافت. در سال ۱۹۶۲، وی به مرتبه استادی ^{۲۷} به استخدام دانشگاه استنفورد درآمد. دکتر مک کارتی در سال ۱۹۶۶ به ریاست آزمایشگاه هوش مصنوعی در دانشگاه استنفورد منصوب شد و اکنون نیز در همان جا به فعالیت مشغول است .



بقیه از صفحه ۱۸

مک کارتی بر روی کامپیوتر پی دی بی ا طراح شده بود عملاً کار خود را آغاز کرده بود. دکتر مک کارتی عضو انجمن علمی در رشته های کامپیوتر و ریاضی است. وی در سال ۱۹۷۱ جایزه معروف تورینگ را از انجمن ماشینهای محاسب ۱۶ دریافت نمود و در سال ۱۹۸۵ توسط انجمن کامپیوتر انستیتوی مهندسان برق و الکترونیک ۷ به عنوان یکی از پیشگامان علم کامپیوتر برگزیده شد.

یادداشتها

*From IEEE Computer Society's booklet describing the 1985 Computer Pioneer Awards.

1. John McCarthy
2. Colifornia Institute of Technology
3. Acting Assistant Professor
4. Dartmouth College
5. Automata Studies, Edited by C.E. Shannon and J. McCarthy, Princeton Univ. Press, 1956.
6. Allen Newell & Herbert Simon
7. IPL= Information Processing Language
8. John Backus
9. Symbolic expressions
10. Fortran List Processing Language
11. Programs with common sense
12. Situation calculus
13. Circumscription
14. Non-monotonic reasoning
15. Timesharing
16. ACM=Association for Computing Machinery
17. IEEE Computer Society.

جمله، دستور العمل، فرمان

در باره یک برنامه اسمبلی می گوئیم که این برنامه تشکیل شده است از چندین دستور العمل اسمبلی.

جمله های یک برنامه مثلاً "به زبان کوپول توسط کامپایلر کوپول و دستور العملهای یک برنامه اسمبلی توسط اسمبلر آن به کدهای ماشین تبدیل می گردند. این کدها توسط کامپیوتری که این برنامه ها روی آن رانده می شوند قابل فهم و اجرا می باشد.

هر کامپیوتری دارای مجموعه ای از دستور العملهای قابل فهم و اجرای خاص خود می باشد. به این مجموعه از دستور العملها، INSTRUCTION SET گفته می شود.

فرمان

این کلمه در موارد زیر مطلق بوده و به کار می رود:

- ۱- در مورد یک زبان سطح پائین مثل اسمبلی. در این موارد این کلمه معنا و مفهوم "دستور العمل" را دارد.
- ۲- دستور کنترل کار^۷
- ۳- هر دستوری که از طریق پایانه کامپیوتر به منظور انجام کاری خاص وارد می شود.
- ۴- هر دستور العمل ماشین^۸ که باعث شروع و یا خاتمه عملی شود، مثل دستور خواندن (READ)، توقف (STOP) و یا وقفه (INTERRUPT).

یادداشتها:

1. Statement
2. Instruction
3. Command
4. Assignment
5. Input/Output
6. Array
7. Job Control
8. Machine Instruction



نرم افزار / زبانهای برنامه نویسی /

سؤال ۱: لطفاً تفاوت میان "جمله"، "دستور العمل" و "فرمان" را بیان نمایید.

جمله

در رابطه با زبانهای برنامه نویسی کلمات "جمله" (بیانیه یا حکم) و "دستور العمل" به طور معادل استفاده می شوند. "جمله" بیشتر در رابطه با زبانهای برنامه نویسی سطح بالا و "دستور العمل" بیشتر در رابطه با زبانهای برنامه نویسی سطح پائین به کار می رود.

در یک برنامه که فرضاً "به زبان فورترن نوشته شده باشد، یک "جمله" می تواند به طور مثال بیانگر هر یک از موارد زیر باشد:

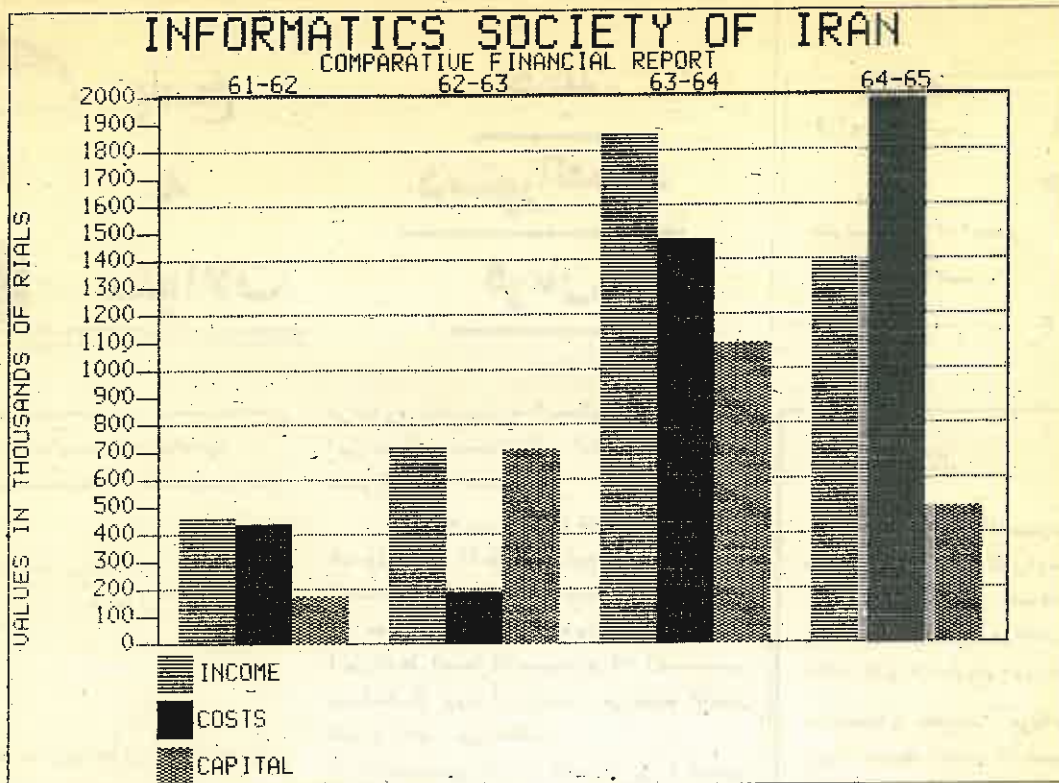
- ۱- یک گمارش^۴
- ۲- یک تعریف برای مشخص کردن نوع یک متغیر
- ۳- یک دستور خواندن و یا نوشتن^۵
- ۴- یک تعریف آرایه^۶

دستور العمل

این لغت در فرهنگ کامپیوتر در موارد زیر متداول می باشد و به کار می رود:

- ۱- زبانهای برنامه نویسی
- ۲- برنامه مبدأ (SOURCE CODE)
- ۳- برنامه ماشین (مقصد) (MACHINE CODE)

در مورد زبانهای برنامه نویسی سطح بالا همان گونه که گفته شد اصطلاح "جمله" بیشتر مرسوم می باشد تا دستور العمل. بنابراین مثلاً "در مورد یک برنامه به زبان کوپول می گوئیم که این برنامه تشکیل شده است از چندین جمله کوپول و یا



بحرانی و نامطلوب بوده است. این مشکل به قدری است که انتشار ماهنامه گزارش کامپیوتر را نیز در خطر قرار داده است و شماره های تیر و مرداد به همین علت به طور یکجا به چاپ رسید. امید است با افزایش حق عضویتها، تمدید به موقع عضویتها و جذب کمکهای مالی بیشتر، وقفه ای در انجام فعالیتها و مختلفانجمن پیش نیاید.

جمع آوری نامه های رسیده به انجمن و تفکیک و توزیع آنها بین مسئولان مربوطه.

چ - امور مالی

همانگونه که در گزارش مالی منعکس گشته است، با توجه به افزایش چشمگیر هزینه ها خصوصاً قیمت کاغذ و مستمر نبودن برخی از درآمدها نظیر دوره های آموزشی و کمکهای مالی، وضع مالی انجمن در سال گذشته بسیار

بقیه گزارش فعالیتها و انجمن... از صفحه ۱۷

متأسفانه فعالیتها انجام گرفته است اکنون موفقیت آمیز نبوده است و انجمن همچنان فاقد دفتر مرکزی میباشد. پاسخ به نامه ها و درخواستهای مشترکین و سایر افراد سازمانها و علاقه مندان به فعالیتها و انجمن.

از ساخت منطقی تمام برنامها، آن را به زبان ماشین ترجمه میکنند و یک نسخه قابل اجرا از برنامها را به زبان ماشین ایجاد میکنند. بحث درباره کامپیوتر و انواع آن و نحوه چگونگی و جزئیات عمل کامپیایل (همگردانی) را به فرستی دیگر موقوف میکنیم.

با داناتها و مراجع:

* - آشنایی با کامپیوتر، دکتر بهروز پرهامی ۱۳۶۳

* - Encyclopedia of Computer Science and Engineering, 1983, PP 51-54

1. Algorithm
2. Basic
3. Programming Languages
4. Statement

بقیه در صفحه ۲۱

تسهیل ارتباط با کامپیوتر، زبانهای سطح بالا^{۱۵} ایجاد شدند. این زبانها به زبان طبیعی نزدیکتر هستند. زبان بیسیک که برنامها را به سبب جذرابا استفاده از آن نوشتیم یک زبان سطح بالاست. به طور کلی زبانهای راکه به زبان ماشین نزدیکتر هستند، زبان سطح پائین^{۱۶} و زبانهای راکه به زبان طبیعی نزدیک هستند، زبان سطح بالایی^{۱۷} میباشند. همانطور که قبلاً گفتیم، برنامهای که به یک زبان سطح بالا نوشته میشود، توسط مترجم به زبان ماشین تبدیل میگردد. مترجمهای زبانهای سطح بالا را میتوان به دو نوع کامپایلر^{۱۸} (همگردانی) و مفسر^{۱۹} تقسیم کرد. مفسرها هر حکم برنامهای راکه به یک زبان سطح بالا نوشته شده است، ترجمه و بلافاصله اجرا میکنند و تنها پس از اتمام ترجمه و اجرای یک حکم، به ترجمه و اجرای حکم بعدی میپردازند. اما کامپایلرها با استفاده

بقیه نقطه شروع مبتدیان (از صفحه ۱۳)

مدتی، برای ماشینهای زبانهای نمادین^{۱۲} بدایع گردیدند. در این زبانها، دنباله صفر و یکها، توسط نویسه های یا دما ندنی نشان داده میشود. دستورالعملی که در زبان ماشین کامپیوتر HP 21 MX نوشتیم، در این زبان به صورت CAX (مخفف COPY A TO X) نوشته میشود. ملاحظه میکنید که به خاطر سپردن و به کار بردن چنین دستورالعملی بسیار آسانتر است. این زبان، زبان اسمبلی^{۱۳} (همگذاری) نامیده میشود.

ایجاد زبان اسمبلی، مستلزم وجود برنامهای مترجمی بود که برنامهای راکه به این زبان نوشته میشوند، به زبان ماشین ترجمه کند. این برنامها، اسمبلر^{۱۴} (همگذار) و عمل ترجمه از زبان اسمبلی به زبان ماشین، اسمبل کردن (همگذاری) نامیده میشود. پس از ایجاد زبانهای اسمبلی، برای



بقیه اخبار انجمن

از صفحه ۲

برگزاری دوره آموزشی "تجزیه و تحلیل سیستمها"

تعیین جوایز خوارزمی سال ۱۳۶۴

دوره آموزشی تجزیه و تحلیل سیستمها از تاریخ ۶۵/۶/۳ تا ۶۵/۶/۱۱ در دانشکده مهندسی صنایع دانشگاه امیرکبیر تهران با حضور یازده شرکت کننده برگزار گردید. در این دوره که توسط آقای ابراهیم نقیب زاده مشایخ تدریس می شد، مفاهیم اساسی تجزیه و تحلیل سیستمهای نرم افزاری و وظایف تحلیلگر سیستم تشریح گردید.

به منظور تسریع آموزش شرکت کنندگان، جزوهای حاوی مطالب ارائه شده در کلاس تهیه شده و در اختیار شرکت کنندگان دوره قرار گرفت. انجمن انفورماتیک ایران از آقایان محمد تقی لوسانی عضو علاقمند انجمن و رئیس مرکز محاسبات دانشگاه امیرکبیر و مهندس نصرالله مرعشی رئیس دانشکده صنایع این دانشگاه که مساعدتهای شایانی جهت تهیه محل برگزاری این دوره مبذول داشتند، تشکر می نماید.

از سوی هیئت داوران برگزیده هیئت اجرایی جوایز خوارزمی برای بهترین مقالات چاپ شده در نشریات گزارش کامپیوتر سال ۱۳۶۴ در دو گروه دانشجویی و آزاد برگزیده شدند. این هیئت همچنین تعدادی از مقالات را نیز به عنوان مقاله های ممتاز معرفی نمود. جوایز برندگان در تاریخ ۲۷ بهمن ۶۵ همزمان با برگزاری مجمع عمومی انجمن در محل وزارت فرهنگ و آموزش عالی واقع در خیابان استاد نجات الهی اهداء خواهد گردید. در این رابطه از سازمانهای مسئول و نیز برخی از اعضای حقوقی انجمن درخواست شده است تا جهت پربارتر کردن جوایز با انجمن همکاری بعمل آورند. مشروح این خبر در همین شماره از نشریه درج شده است.

کمک مالی وزارت فرهنگ و آموزش عالی

از سوی معاونت پژوهشی وزارت فرهنگ و آموزش عالی مبلغ یکصد هزار ریال به عنوان کمک مالی در اختیار انجمن انفورماتیک ایران قرار گرفت. انتظار می رود از این پس وزارت فرهنگ و آموزش عالی سهم بیشتری در کمک به فعالیتهای علمی انجمن به عهده گیرد.

برنامه ریزی چند دوره آموزشی

از سوی کمیته آموزش انجمن، تعدادی دوره آموزشی در زمینه های مختلف علوم کامپیوتر تدوین گردیده و است که تا پایان سال ۱۳۶۵ به اجرا در خواهند آمد. مشخصات دوره ها و تاریخ برگزاری آنها متعاقباً اعلام خواهد شد.

قرار گرفت و پس از اطمینان از صحت آنها و حذف قسمت پرفرا شده برگها (حاوی شماره عضویت) قرائت آنها به عمل آمد. جمعا ۱۰۵ رأی به انجمن رسیده بود که پس از قرائت، نتایج زیر حاصل گردید:

- ۱- آقای بهزاد روحانی ۷۹ رأی
- ۲- آقای حمید خورشید رحیم زاده ۶۳ رأی
- ۳- خانم میترا آل یاسین ۶۲ رأی
- ۴- خانم میترا کا تو زیان ۵۷ رأی
- ۵- خانم پروین کوچمشکی ۴۴ رأی
- ۶- آقای علی اکبر نهرودی ۳۸ رأی

بدین ترتیب دونفر اول به عنوان اعضای اصلی و دونفر بعد به عنوان اعضای علی البدل هیئت اجرایی برگزیده شدند.

سخنرانی علمی

همزمان با برگزاری مجمع عمومی انجمن در تاریخ هفتم مهرماه، آقای حسن کا تو زیان تحت عنوان "آدمک مصنوعی پارس ۱" که ساخته خود ایشان است سخنرانی کردند. در پایان این سخنرانی، آیین آدمک و قابلیت های آن به نمایش گذاشته شد.

بقیه نقطه شروع مبتدیان از صفحه ۲۰

14. Assembler
15. High Level Language
16. Low Level Language
17. Compiler
18. Interpreter

9. Structures
10. Structured Programming
11. Translators
12. Symbolic
13. Assembly

5. Loop
6. Count Loops
7. PL/I
8. While Loops

بقیه از صفحه ۲

پس از بررسی نهایی، شش مقاله در دو گروه دانشجویی و عادی امتیازی بالاتر از ۷۵ آورده و به این ترتیب از میان آنان برندگان جوایز خوارزمی مشخص گردیدند. یادآوری این نکته ضروری است که در ارزیابی، اصل مقالات قبل از ویرایش بلامک قرار گرفتند و نه متن چاپ شده آنان.

در گروه آزاد:

مقاله "پیشرفتهای در زمینه معماری کامپیوتر بر اساس جریان داده ها" ترجمه خانم میترا آل یاسین

در گروه دانشجویی:

جایزه این گروه مشترکا "به دو مقاله زیر تعلق گرفت:

مقاله "گامی در جهت افزایش توان استنتاج در کامپیوترها"

گردآوری و تألیف آقای حسین کمالی

مقاله "نسل ششم کامپیوتر یا کیپی برداری از مغز"

ترجمه آقای حسن کا تو زیان

هیئت داوران سه مقاله دیگر را نیز به عنوان مقالات برگزیده و ممتاز معرفی نمود:

مقدمه ای بر طراحی به کمک کامپیوتر

تألیف آقای علی پارسا

زبانهای مبتنی بر جریان داده ها

ترجمه خانم میترا آل یاسین

گامی نو در برقراری ارتباط بین انسان و ماشین

ترجمه آقای حسن کا تو زیان

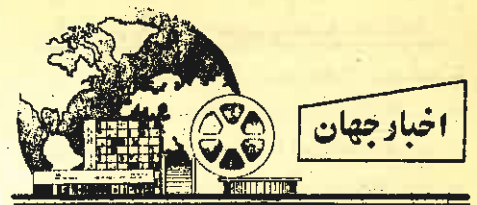
یک راه حل ساده برای مرتب سازی کامپیوتری اطلاعات فارسی

تألیف: قاسم جابری پور

جوایز برندگان، طی مراسمی همزمان با برگزاری رسمی هفتمین مجمع عمومی سالانه انجمن به آنان اهداء خواهد شد.

کمیته انتشارات

انجمن انفورماتیک ایران



کارت رتبیج جدید خود را که کاملاً همساز با آی بی ام ایجا دگر دیده است عرضه داشت. شرکت دیتا سیستمز^۳، خانواده ابرریزرها را که کامپیوترهای^۴ ۳۲ بیتی خود با قدرت پردازش ۱ میلیون دستورالعمل در ثانیه را که از زیر پردازنده‌های ۶۸۰۲۰ و ۶۸۰۰۰ موتورولا - استفاده می‌کنند به معرض دید عموم گذارد. قیمت این ماشینها از ۸۰۰۰ الی ۳۵۰۰۰ دلار برآورد شده بود، ماشینهای مزبور تحت سیستم عامل یونیکس عمل می‌نمایند.

شرکت توشیبا کامپیوتر قابل حمل جدیدی را با نام T1100 عرضه داشت که مجهز به دیسک لرزان (فلایی) ۳ ۱/۲ اینچ با ظرفیت ۷۲۰ کیلوبایت می‌باشد، قیمت این ماشین ۲۰۰۰ دلار اعلام شده بود.

شرکت نیک (NEC) نیز ریزرکا کامپیوتر ۳۲ بیتی خود با نام Astra را معرفی نمود که تحت سیستم عامل یونیکس عمل می‌کند. ریزپردازنده ۲۰، موتورولا ۶۸۰۲۰ بوده و دارای ۱۶ کیلوبایت حافظه سریع^۵ می‌باشد. در مجموع، این نمایشگاه‌ها هادارائه محصولات متنوع و جدید با قیمت‌های نسوزول یافته بود.

به نقل از:

COMPUTER WORLD

June 23, 1986

1. National Computer Conference (NCC)
2. Memorex
3. Data Systems
4. Super Microcomputers
5. Cache

برنامه‌های سودمند برای برنامه‌های DBASE

اخیراً شرکت Microproducts Gryphon، مجموعه‌ای از برنامه‌های سودمند را به بازار عرضه کرده است که قابلیت حافظه DBASE را از ۶۳ متغیر به بیش از ۸ میلیون متغیر افزایش می‌دهد. این بسته نرم‌افزاری که DB/RA نام دارد شامل ۲۲ فرمان است که می‌توانند مستقیماً از طریق DBASE فراخوانده شوند. با استفاده از این برنامه‌ها می‌توان برنامه‌ها را به مکان می‌بند تا با پیغام داده‌ها به عنوان آرایه^۱ در حافظه را بکنند. این بسته نرم‌افزاری دارای فرمان‌های^۲ جهت جستجوی جدا و جستجوهای محدودده^۳ جداگانه^۴ ۱۰۰ آرایه است و به علاوه امکانات دیگری از قبیل کارکرد پیچیده^۴ را نیز دارا می‌باشد. قیمت این نرم‌افزار ۲۰۰ دلار تعیین شده است.

به نقل از: Byte, March 1986

1. Utililites
2. array
3. range search
4. windowing

های شخصی ۱۸ درصد (تا ۱۲ میلیارد دلار) افزایش خواهد یافت. در سال ۱۹۸۵ صادرات کامپیوترهای شخصی تا ۲۰ درصد افزایش را نشان می‌دهد. بیش بینی می‌شود در سال جاری درآمد نرم‌افزار کامپیوترهای شخصی ۳۷ درصد افزایش یا بدو به مرز ۳/۷ میلیارد دلار برسد.

به نقل از:

Mini-Micro Systems

March 1986

اولین شبکه ارتباطی داده‌ها در هند

شرکت COMPUTER MAINTENANCE در حال برنامه‌ریزی ایجاد اولین شبکه ارتباطی اطلاعات در هندوستان است. در بدو امر این شبکه مراکز بزرگ کامپیوتری پنج شهر را به یکدیگر متصل می‌کند. این شبکه که Indonet نام دارد از تکنولوژی Packet Switching و استاندارد قرار داده ارتباطی آی بی ام برای معماری شبکه‌ها استفاده می‌نماید.

اولین مرحله این برنامه قبل از پایان سال ۱۹۸۵ تکمیل شده است. استفاده کنندگان راه دور با در اختیار داشتن پایانه‌ها و چاپگرهای ASCII از امکاناتی چون توسعه برنامه‌ها، ایجاد دو ویرایش پرونده‌های مبدأ، ترجمه، اتصال - ذخیره برنامه‌ها، پردازش، تهیه گزارش، وارد کردن داده‌ها به صورت درون خطی^۱ و دستیابی به دیگر سیستم‌های کامپیوتری موجود در شبکه جهت پردازش توزیعی داده‌ها^۲ برخوردارند. در خطوط محلی این شبکه، داده‌ها با

سرعت حداکثر ۱۲۰۰ بیت در ثانیه به صورت کاملاً دوطرفه انتقال می‌یابند.

به نقل از:

Data Processing

Jan./Feb. 1986

1. Source files
2. Online data entry
3. distributed data Processing
4. full duplex

نمایشگاه NCC

کنفرانس ملی کامپیوتر در نمایشگاه امسال خود معروف محصولات جدیدی بود. در این نمایشگاه، فقط شرکت آی بی ام به تنهایی ۱۲۵ نوع محصول خود را به معرض نمایش گذاشته بود. شرکت ممورکس^۱ نیز توان

هیولیت پکارد بسته نرم‌افزاری نگاره‌ای

شرکت هیولیت پکارد اقدام به ایجاد یک بسته نرم‌افزاری نگاره‌ای (گرافیکی) نموده است که بر روی ریزرکا کامپیوترهای وکترال^۱ (ساخت این شرکت) آی بی ام همسازان آن قابل اجرا می‌باشد.

این بسته که Graphics Gallery نام دارد، خود از دو برنامه به نام‌های Drawing و Gallery و Charting Gallery تشکیل می‌گردد که به ترتیب، اولی به منظور ایجاد، ترسیم و اصلاح نمودارهای تجاری، سازمانی و روندنماها و دومی در جهت ایجاد نمودارهای آماری (ستونی، خطی، دایره‌ای و...) - طراحی گردیده‌اند. در این بسته امکان اضافه نمودن متون مختلف در نمودارها و اشکال گوناگون به نمودارهای رسم شده موجود می‌باشد.

این نمودارها با نمودارهای رسم شده به وسیله بسته‌های لوتوس^۲ - ۱ - ۲ و سمفونی کاملاً همساز بوده و می‌توانند تلفیق گردند.

قیمت این دو برنامه مجموعاً در حدود ۷۰۰ دلار می‌باشد.

به نقل از:

Computer World

July 6, 1986

۱. Vectra

پیش‌بینی صادرات کامپیوتر در سال ۱۹۸۶

بر اساس یک برآورد، میزان صادرات تجهیزات کامپیوتری آمریکا در سال ۱۹۸۶ نسبت به ۱۹۸۵، ۱۶ درصد افزایش یافته و به مبلغ تقریبی ۶۲ میلیارد دلار خواهد رسید. اما بسیاری از این صادرات به دلیل وجود انبارهای بزرگ و عدم رشد قابل توجه اشتغال کارگران خط تولید کامپیوتر، خارج از رده خواهند بود.

بر اساس همین برآورد، صادرات ابر کامپیوترها (سوپر کامپیوترها) در سال ۱۹۸۶ به میزان ۱۸ درصد (تا ۱۴ میلیارد دلار)، نرم‌افزار کامپیوترهای کوچک (مینی) ۲۸ درصد (تا ۸/۷ میلیارد دلار) و کامپیوتر-

د	منصوری شیرازی نصرالله	۵۵۴۸
د	دانشی کونکائی احد	۵۶۶۲
ع	مهیا به محمد	۱۶۲۳
ع	اسدی رحیم	۱۲۸۰

تمدید عضویت
طی ماههای تیر، مرداد و شهریور،
افراد زیر عضویت خود را در انجمن تمدید
کرده اند:

د	عسکری صباغی	۵۶۸۲
ع	سماواتی فرداد	۱۵۱۵

د	مسعود راهور	۵۱۷۳
ع	محمدرضا مر	۱۶۷۹
د	امیر خاللی سیدعلیرضا	۵۴۲۳
ع	حداد شکری التفات (۶۴-۶۵)	۱۱۱۲
ع	اسدی مقدم ایران دخت	۱۱۱۳
و	قلعه گلایی علی	۳۰۵۲
د	سرداری ایروانی عنایت الله (۶۴-۶۵)	۵۴۷۰
د	مقصودی شاهین	۵۴۹۲

ع	امیری دوان منصور (۶۴-۶۵)	۱۵۷۵
ع	افشاری علی آباد سفندیار (۶۴-۶۵)	۱۰۱۹
ع	بیات غلامرضا (۶۴-۶۵)	۱۰۸۴
ع	نبیثی مجتبی	۱۴۳۷
ع	جوان داریوش	۱۱۱۸

ع	مجتهدزاده مهری	۱۴۲۷
ع	لری امینی محمد	۱۶۶۰
ع	غفاریان خیابزاده احمد	۱۴۲۴
د	هوشمند فرهاد	۵۲۵۲

د	فرزاد فرامرز	۵۲۵۱
د	کتانچی مازیار	۵۶۲۸
د	محمدنژاد غلامرضا	۵۵۸۹
د	آشتی آشتیانی ویولت	۵۶۰۰
د	حنیفی افشار شرمین	۵۵۹۹
د	بنگدار علیرضا	۵۶۴۰

ع	کسرائی محمدیوسف	۱۵۵۲
ع	ثابتان محمدعلی	۱۶۳۸
د	کامفر آرمینا	۵۴۹۹
د	سمسارزاده شهلا (۶۴-۶۵)	۵۲۹۵
ع	روحانی رانگویی محمدتقی	۱۰۵۱
د	ظاهریان احد	۵۵۰۸
د	همدانیان پور مهرداد	۵۷۰۴



اعضاء

موعد تمدید عضویتها

با توجه به شروع سال جدید عضویت،
لزام است کلیه اعضای انجمن، عضویت خود را
برای سال ۶۶-۶۵ تمدید نمایند. حق عضویت
اعضای عادی و وابسته ۲۵۰۰ ریال، اعضای
دانشجویی ۱۰۰۰ ریال و اعضای حقوقی حداقل
۱۵۰۰۰ ریال میباشد.
در ضمن اعضای که تاکنون عضویت خود
برای سال ۶۵-۶۴ را تمدید نکرده اند، باید
علاوه بر مبلغ فوق، حق عضویت سال گذشته
را نیز (اعضای عادی و وابسته ۱۵۰۰ ریال
و اعضای دانشجویی ۵۰۰ ریال) پرداخت نمایند.

اعضای جدید

طی ماههای تیر، مرداد و شهریور، افراد
زیر به عضویت انجمن درآمده اند:

د	سجودی پاشته آ	۵۷۶۷
د	حیدرزاده هوشنگ	۵۷۶۸
ع	بادگاری عباس	۱۷۰۸
د	نوری محسن	۵۷۶۹
د	فتحی خلفیادام مهدی	۵۷۷۰
د	چریری فریدون	۵۷۷۱
ع	بهمن دزی بهمن	۱۷۰۹
د	نصیری رانی مسعود	۵۷۷۲
د	صامیان صدیقه	۵۷۷۳
د	محمدزاده مجید	۵۷۷۴
د	پولادی دریش مهران	۵۷۷۵
ع	وزیری ساسان	۱۷۱۲
و	افتخارحسینی سیداحمد	۳۲۲۲

بدین ترتیب تا پایان شهریور، ۲۱ اعضای
انجمن به قرار زیر است:

عادی وابسته	دانشجویی برجسته	حقوقی جمع کل
۲۵۵	۱۲۷	۵۷۶
۱	۱۱	۱۰۷۰

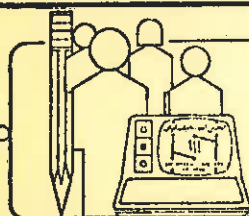
نشریات برگشتی

نشریات افراد زیر به انجمن برگشت
داده می شوند. از این افراد دوباره دیگر اعضای
انجمن خواهند شد. این نشریات در صورت اطلاع،
نشانیهای صحیح ایشان را در اختیار انجمن
قرار دهند.

آقای مهرداد صفائی	(۵۵۳۷)
آقای محسن مهرش	(۵۷۲۹)
آقای بهرام فرخی	(۵۵۳۶)
آقای جواد مرما	(۵۳۰۴)
آقای آلبرت و سکا نیان	(۵۳۳۰)
آقای بهمن جعفری دینانی	(۳۱۴۵)
آقای عبدالحسن دمستانی (۱۴۵۵) از آمریکا	
آقای کامران فاتح نژاد	(۵۶۲۵)
آقای کریم الله بصیری	(۱۶۵۶)
آقای احمد میرعلائی	(۱۵۶۵)
آقای لطفعلی بخشی	(۱۰۰۰)

تصحیح نام

در شماره های گذشته در ستون اعضا، نام
خانم گوهر تاج ارشد منصب به اشتباه خاتم
گوهر تاج درج گردیده بود که بدین وسیله
تصحیح میگردد.



گزارش کامپیوتر

ماهنامه انجمن انفورماتیک ایران

زیر نظر کمیته انتشارات انجمن

نشانی: صندوق پستی ۱۱۹۶ - ۱۴۱۵۵، تهران، ایران
سرپرست کمیته انتشارات و سردبیر: حمید رضا رهبر

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

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

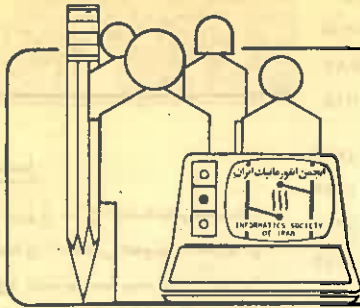
Address: P.O.Box 14155-1196, Tehran, Iran-Editor: HAMID REZA RAHBAR

حق عضویت سالانه اعضا: (عادی ۴۵۰۰ ریال
(شامل اشتراک ماهنامه) وابسته ۲۵۰۰
(دانشجویی ۱۰۰۰)
حق اشتراک یک ساله گزارش کامپیوتر ۲۰۰۰
هزینه پست برای اعضا با مشترکین
مقیم خارج از کشور ۱۰۰۰

لطفاً رسید پرداخت وجه به حساب جاری
شماره ۱۱۱۷ انجمن نزد بانک صادرات شعبه
دانشگاه صنعتی شریف، تهران (شعبه شماره ۹)
(۱۶۵۲) را به درخواست خود ضمیمه کنید.
پرداخت وجه از طریق هر یک از شعب بانک
صادرات در تهران و شهرستانها ممکن است.
برای دریافت بروشور حاوی فعالیت های
انجمن و تقاضا نامه عضویت، با ما
مکاتبه کنید.

منتشر شد

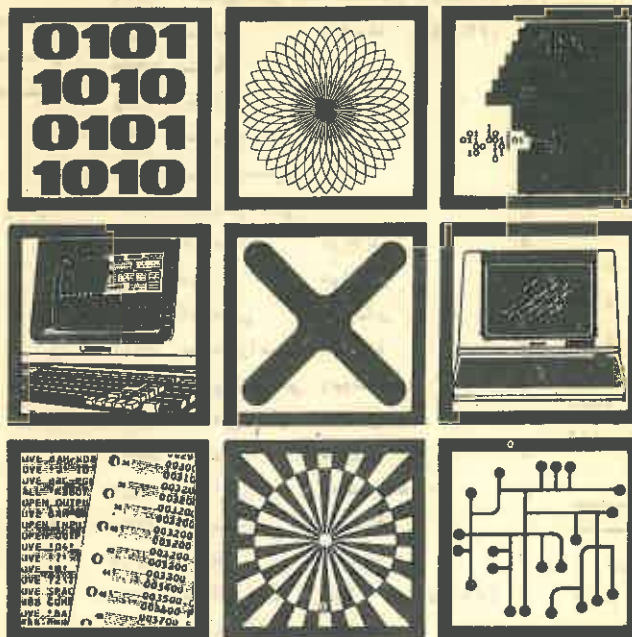
سالنامه سال ۱۳۶۴ در ۲۸۸ صفحه آماده فروش است



گزارش کامپیوتر

ماهنامه انجمن انفورماتیک ایران

۱۳۶۴



سال هفتم - شماره های ۱ - ۱۲

(شماره های پیاپی ۶۳ - ۷۴)

سالنامه سال ۶۴ و دیگر نشریات انجمن را می توانید

از طریق مکاتبه با انجمن دریافت دارید .

قیمت برای اعضا : ۱۰۰۰ ریال

قیمت برای غیرعضو : ۱۵۰۰ ریال