

5697

السلامة للجميع

۱۰۰۰
۱

۲۴

نویس

www.ketab.ir



بازی صفر و یک

زهرا فرشچی

سرشناسه: فرشچی، زهرا، ۱۳۵۳ -
عنوان و نام پدیدآور: بازی صفر و یک / نویسنده: زهرا فرشچی
مشخصات نشر: قم: بوکتاب، ۱۳۹۷
مشخصات ظاهری: ۲۴۰ ص: ۱۴/۵×۲۱/۵ س.م.
شابک: ۹۷۸-۶۰۰-۹۹۷۸۳-۶-۶
وضعیت فهرست نویسی: فیبا
موضوع: برنامه نویسی
Computer programming
موضوع: زبان های برنامه نویسی کامپیوتر
Electronic computers -- Programming languages
ردمبندی کنگره: ۱۳۹۷ ب۲ ق۳/۶/۴۸۷۶
ردمبندی دیویی: ۰۰۵/۱
شماره کتابشناسی ملی: ۵۶۰۱۰۵۰

انتشارات بوکتاب

بازی صفر و یک

نویسنده: زهرا فرشچی

ویراست: حسین طبع

طرح جلد: جواد

تاریخ انتشار: ۱۳۹۷ - نوبت چاپ: اول

شمارگان: ۱۰۰۰ نسخه

قیمت: ۳۵۰۰۰ تومان

شابک: ۹۷۸-۶۰۰-۹۹۷۸۳-۶-۶

تهران: میدان انقلاب، خیابان شهدای زاندارمری، پلاک: ۵۸.

طبعه اول. تلفن: ۰۲۱۶۶۴۸۴۱۷۹.

کلیه حقوق برای ناشر محفوظ است.

فهرست

۱۵	مقدمه
۱۷	تاریخچه
۱۸	مفتر بیسیک
۲۲	نگاه اولیه به QBASIC
۲۲	صفحه اصلی
۲۵	استفاده از منوهای QBASIC
۲۷	ایجاد و اجرای یک برنامه
۲۹	وارد کردن یک برنامه QBASIC
۳۲	:PRINT i
۳۳	معرفی زبان‌های برنامه‌نویسی
۴۰	معرفی BASIC
۴۱	پیش زمینه
۴۳	سالهای نخست: دوره مینی کامپیوترها
۴۶	رشد خیره کننده: دوره کامپیوترهای خانگی
۴۹	بلوغ: دوره کامپیوترهای شخصی
۵۳	معرفی زبان C
۵۷	مراحل اجرای یک برنامه C
۵۹	۸ ویژگی برنامه نویسی C
۶۱	۲. کلمات کلیدی کم
۶۲	۳. عدم وجود رشته‌های صریح

۶۳	۴. دست کاری اشاره گر
۶۴	۵. قابلیت تعویض محل اشاره گر ها و آرایه ها
۶۴	۶. استفاده گسترده از ماکروهای «define»
۶۵	۷. مژول های کپسوله شده در فایل ها
۶۵	۸. کتابخانه های خارجی
۶۶	زبان برنامه نویسی C#
۶۷	شی گرایی به چه معنا است؟
۶۹	C# برای من مناسب است؟
۷۰	با C# چه نرم افزارهایی را می توانم توسعه دهم؟
۷۱	Net. چه ارتباطی با C# دارد؟
۷۱	C# و .net چیست و آیا مناسب من است؟
۷۲	جاوا
۷۴	تاریخچه زبان برنامه نویسی Java
۷۷	اصول اولیه زبان برنامه نویسی Java
۷۸	کاربردهای زبان برنامه نویسی Java
۷۹	پیاده سازی ها
۸۱	عملکرد
۸۲	مدیریت حافظه خودکار
۸۵	قاعده نحوی
۸۹	جاوا اسکریپت
۹۱	تاریخچه
۹۳	ویژگی ها
۹۳	زبان امری و ساخت یافته

۹۳	پویایی
۹۴	تابعی بودن
۹۵	ساختار شماتیک» محوری
۹۶	امکانات دیگر
۹۶	Literal های آرایه و شی
۹۷	عبارات منظم
۹۷	سی پلاس پلاس
۹۷	موارد استفاده از زبان سی پلاس پلاس
۱۰۰	پی اچ پی
۱۰۲	تاریخچه
۱۰۶	پرل (PERL)
۱۰۶	تاریخچه
۱۰۹	پرل (PERL)
۱۱۲	پونی
۱۱۳	نام
۱۱۴	نماد شتر
۱۱۴	نماد پیاز
۱۱۴	ویژگی ها
۱۱۵	طراحی
۱۱۶	پیاده سازی
۱۱۹	دسترسی
۱۲۰	ویندوز
۱۲۱	پایگاه داده

۱۲۳		بهینه‌سازی
۱۲۴		پرل ۶
۱۲۶		قابلیت‌ها
۱۲۷		کاربرد پرل
۱۲۷		پرل به عنوان یک زبان چسبنده (Glue Language)
۱۲۷		پرل به عنوان یک زبان کراس پلتفرم (Cross Platform)
۱۲۸		رل و CGI
۱۲۸		سوئیفت
۱۳۰		تاریخچه
۱۳۱		سی اس اس
۱۳۲		استانداردهای تازه
۱۳۳		برتری‌ها
۱۳۳		پهنای باند
۱۳۴		نسخه‌ها
۱۳۴		اس کیو ال
۱۳۵		پایگاه داده رابطه‌ای (Relational Database)
۱۳۸		ویرایش اینترنت‌رایز
۱۳۸		استاندارد
۱۳۹		وب
۱۳۹		Workgroup
۱۴۰		Express
۱۴۱		آژور دیتابیس
۱۴۱		آژور

۱۴۱	 (Compact) SQL CE
۱۴۲	 Developer
۱۴۲	 (Embedded) SSEE
۱۴۲	 Evaluation
۱۴۳	 Fast Track
۱۴۳	 LocalDB
۱۴۳	 (Analytics Platform System (APS
۱۴۴	 Data warehouse Appliance Edition
۱۴۴	 MSDE
۱۴۴	 ویرایش شخصی
۱۴۵	 دیتاستر
۱۴۵	 چرا از دیتابیس استفاده کنیم ؟
۱۴۸	 چرا از دیتابیس استفاده کنیم ؟
۱۴۸	 سیستم مدیریت
۱۴۹	 نتیجه گیری
۱۴۹	 جی کوئری
۱۵۰	 ویژگی ها
۱۵۰	 اچ تی ام ال (HTML)
۱۵۲	 آشنایی با برخی تگ های html
۱۵۲	 مای اسکيوال
۱۵۶	 اليکسیر
۱۵۸	 توسعه پذیری و DSL ها
۱۵۸	 اکو سیستم در حال رشد

۱۵۹.....	Erlang با سازگاری
۱۵۹.....	راست
۱۵۹.....	اهداف اولیه طراحی زبان Rust
۱۶۰.....	Rust از چه زبان‌ها و سیستم‌هایی الهام گرفته است؟
۱۶۱.....	اسکالا
۱۶۲.....	آشنایی دلایل محبوبیت زبان اسکالا
۱۶۳.....	آر (R)
۱۶۳.....	تعریف زبان R
۱۶۳.....	تاریخچه زبان R
۱۶۴.....	طراحی این زبان
۱۶۵.....	ویژگی‌های زبان R
۱۶۶.....	محبوبیت زبان برنامه‌نویسی R
۱۶۷.....	نیم نگاهی به امکانات R
۱۶۸.....	روبی
۱۷۱.....	کاربرد زبان برنامه‌نویسی روبی
۱۷۱.....	برخی از امکانات Ruby
۱۷۲.....	جمع بندی کلی از برنامه نویسی روبی
۱۷۴.....	پاسکال
۱۷۵.....	تاریخچه
۱۷۶.....	توربو پاسکال
۱۷۸.....	پاسکال دلفی
۱۷۸.....	چکیده
۱۷۹.....	پیاده‌سازی

۱۸۶.....	ساختارهای زبان
۱۸۶.....	انواع داده.....
۱۸۷.....	کامپایلرها و مفسرها
۱۹۳.....	استانداردها
۱۹۵.....	دسته‌بندی‌ها
۱۹۷.....	انتقادات
۱۹۹.....	مزایا و معایب زبان برنامه‌نویسی پاسکال
۲۰۱.....	اجزای تشکیل دهنده یک برنامه
۲۰۳.....	ساختار کلی برنامه در زبان پاسکال
۲۰۵.....	تورینگ.....
۲۰۶.....	پیاده سازی بار.....
۲۰۶.....	Tplus
۲۰۷.....	OpenT
۲۰۸.....	مهارت‌های برنامه نویسی اندروید.....
۲۰۸.....	۱- با ساختار داخلی فریم‌ورک اندروید بیشتر آشنا شوید
۲۰۹.....	۲- ترس از «دست دادن» را کنار بگذارید.
۲۰۹.....	۳- تا می‌توانید کدهای بیشتری بخوانید
۲۱۰.....	۴- زبان‌های بیشتری یاد بگیرید
۲۱۰.....	۵- الگوهای طراحی جاوا را یاد بگیرید
۲۱۱.....	۶- در پروژه‌های متن‌باز مشارکت کنید
۲۱۲.....	۸- برنامه‌ی خود را بر مبنای یک معماری درست بنا کنید
۲۱۲.....	۹- کدنویسی تمیز را بیاموزید

- ۱۰- زمانی را برای یادگیری بهترین تمرین‌های برنامه نویسی اندروید صرف کنید..... ۲۱۲
- ۱۱- از زمان خالی خود برای شنیدن پادکست استفاده کنید..... ۲۱۳
- ۱۲- واقع نگر باشید؛ بیش از حد مهندسی نکنید..... ۲۱۳
- ۱۳- در مورد طراحی هم مطالعه کنید..... ۲۱۴
- ۱۴- کمال‌گرا باشید..... ۲۱۴
- ۱۵- تداوم و پشتکار، کلید موفقیت شماسست..... ۲۱۵
- ۱۶- قدم‌های کوچک بردارید، آهسته و پیوسته حرکت کنید..... ۲۱۵
- ۱۷- همیشه یک پروژه تفریحی داشته باشید..... ۲۱۶
- ۱۸- تست‌های بیشتری بنویسید..... ۲۱۶
- ۱۹- خودتان را به TDD عادت دهید..... ۲۱۷
- ۲۰- یک مکانیزم اتوماسیون مناسب برای نسخه‌های مختلف برنامه تدوین کنید..... ۲۱۸
- ۲۱- نگاهی هم به رویکرد برنامه نویسی واکنش‌گرا داشته باشید..... ۲۱۸
- ۲۲- استفاده از Kotlin برای توسعه اندروید را فرا بگیرید..... ۲۱۸
- ۲۳- ارتباط فعالی با دیگر توسعه‌دهندگان برقرار کنید..... ۲۱۹
- ۲۴- به استفاده از کلیدهای میانبر صفحه کلید عادت کنید..... ۲۱۹
- ۲۵- سعی کنید هر هفته حداقل یک چیز جدید در مورد اندروید یاد بگیرید..... ۲۲۰
- ۲۶- هر چیزی که زمانتان را می‌بلعد خودکار کنید..... ۲۲۰
- ۲۷- از دو نسخه اندروید استودیو استفاده کنید..... ۲۲۱
- ۲۸- هر از گاهی تمامی کتابخانه‌ها را بررسی کنید..... ۲۲۱

- ۲۹- برای ریفکتور کردن کدهای قدیمی از روش‌های جدید استفاده کنید ۲۲۱
- ۳۰- همیشه کد را بر روی دیوایس‌های رده‌پایین تست کنید ۲۲۲
- ۳۱- بهترین سیستم را خریداری کنید ۲۲۲
- هفت دلیل رشد نکردن برنامه نویسی در ایران ۲۲۳
- ۱- مدارس ۲۲۳
- ۲- منابع فارسی محدود و قدیمی! ۲۲۴
- ۳- رعایت نکردن حقوق برنامه نویس ۲۲۴
- ۴- کمی برداری در روز روشن! ۲۲۵
- ۵- فرینسری ۲۲۵
- ۶- ما اهل اوپن سورس و نرم افزار آزاد نیستیم! ۲۲۶
- ۷- تحریم ۲۲۶
- ۱۵ ویژگی یک برنامه نویس حرفه ای ۲۲۷
- ۱- مهارت‌های تکنیکی موثر ۲۲۷
- ۲- تمایل به یادگیری ۲۲۸
- ۳- مهارت‌های اشکال زدایی ۲۲۹
- ۴- خود را با محیط کار وفق دهد ۲۲۹
- ۵- توانایی حل مشکل ۲۳۰
- ۶- شور و شوق برای کار ۲۳۰
- ۷- آرامش داشتن ۲۳۱
- ۸- مهارت‌های اجتماعی ۲۳۱
- ۹- تنبلی ۲۳۲
- ۱۰- چشم انداز کسب و کار ۲۳۲
- ۱۱- توانایی در برنامه ریزی ۲۳۳

۲۳۳..... ۱۲- توانایی مدیریت شکست

۲۳۳..... ۱۳- ذهنیت کار گروهی

۲۳۴..... ۱۴- تمایل به پژوهش

۲۳۴..... ۱۵- احترام به مهلت داده شده (DeadLine)

۲۳۴..... چگونه برنامه نویس خوبی باشیم

www.ketab.ir

مقدمه

اگرچه برنامه نویسی یک اختراع جدید محسوب می شود اما این در حالی است که ایده ی مجبور کردن یک ماشین یا دستگاه برای انجام یکسری دستورات خاص به سالها پیش - چیزی در حدود یک قرن پیش - باز می گردد. در واقع، یکی از ابتدایی ترین طرح ها برای یک ماشین قابل برنامه ریزی - یا همان کامپیوتر - از کسی به نام Charles Babbage در سال ۱۸۳۴ گفته شد.

در واقع وی پیشنهاد ساخت یک ماشین بخار که به صورت مکانیکی یکسری کارها را انجام دهد - که تحت عنوان موتور تحلیلیگر شناخته می شود - را داده بود. برخلاف سیستم های محاسبه ی آن روز که تنها می توانستند یک عمل محاسباتی را انجام دهند، موتور تحلیلیگر آقای چارلز بابج چندین کار همزمان را می توانست انجام دهد. دستوراتی که به ماشین داده می شد از طریق یکسری کارت ها انجام می پذیرفت که با تغییر تعداد این کارت ها و نوع آن ها، هر کسی می توانست دستورات مد نظر خود را به ماشین برای محاسبه ی آن ها بدهد.

ایده‌ی ماشین‌های قابل برنامه‌ریزی توجه خانمی به نام Ada Lovelace را به خود جلب کرد که یک ریاضیدان بود. لذا او برنامه‌ای نوشت -یا بهتر است بگوییم کارت‌هایی ترتیب داد- که تعدادی عدد خاص را پشت سر هم چاپ می‌کرد (با توجه به این که این خانم اولین کسی بود که کار خاصی را با Analytical Engine یا همان موتور تحلیلگر انجام داد، وی را می‌توان به نوعی اولین برنامه‌نویس دنیا نیز تلقی کرد که نامگذاری زبان برنامه‌نویسی Ada هم به احترام وی صورت گرفته است).

چیزی پس از گذشت یک صد سال، اولین کامپیوتر به معنای واقعی کلمه در سال ۱۹۴۴ توسط ارتش ایالات متحده طراحی شد. این ماشین که Electronic Numerical Integrator and Computer یا به اختصار ENIAC نامیده شد، از کابل‌ها و سویچ‌های مختلفی تشکیل شده بود و برای دستور دادن به آن، برنامه‌نویس می‌بایست جای کابل‌ها و سویچ‌ها را تغییر دهد (جالب است بدانیم که اولین برنامه‌نویسان این کامپیوتر همگی خانم بوده‌اند!) نکته‌ی اصلی اینجا است که کامپیوترهای آن موقع بسیار حجیم و گران بودند و کار کردن با آن‌ها واقعا تخصص نیاز داشت!