

خطای هزاره و نقص نرم افزارهای مبتنی بر تاریخ هجری شمسی

رضا روشنی

موسسه آموزش عالی غیر انتفاعی، غیر دولتی کمیل کردکوی
roshani@komeil.ac.ir

خلاصه

در آستانه شروع قرن جدید میلادی و نزدیک شدن به سال ۲۰۰۰ میلادی یکی از بزرگترین بحران‌ها و دغدغه‌های دنیای نرم افزار شکل گرفت. این مشکل آنچنان بزرگ بود که کشورهای جهان بودجه‌های چندین میلیون دلاری جهت بررسی و رفع مشکل اختصاص دادند. تاریخ هجری شمسی تا چند سال دیگر وارد قرن جدید، یعنی سال ۱۴۰۰ خواهد شد. در بسیاری از نرم افزارهای فارسی و بانک‌های اطلاعاتی برای جستجوی آسان‌تر و عدم تبدیل مکرر تاریخ میلادی به شمسی و بالعکس، سال مربوط به تاریخ شمسی به صورت خلاصه دو رقمی ذخیره می‌شود. این مسئله مشکلاتی را در این نرم افزارها بوجود می‌آورد که ما در این مقاله سعی به بررسی آن‌ها و سپس ارائه پیشنهاداتی در رابطه با حل این مشکلات می‌پردازیم.

کلمات کلیدی: خطای هزاره شمسی، نقص نرم افزاری، نرم افزارهای بومی، تاریخ هجری شمسی

۱. مقدمه

در حال حاضر تقویم رسمی کشورهای ایران و افغانستان فقط با تفاوت اسامی ماه‌ها از هر جهت گاهشماری یکسان به حساب می‌آیند. در ایران از نامهای فارسی میانه ماه‌ها و در افغانستان از نام‌های عربی صورت‌های فلکی منطقه البروج استفاده می‌شود.

در کشور ایران گاهشماری هجری خورشیدی، بر اساس مبانی گاهشماری جلالی و پس از تغییراتی در گاهشماری جلالی در قالب گاهشماری هجری خورشیدی برجی در سال ۱۲۸۹ خورشیدی برای امور مالی کشور رسمیت یافت. سپس با تغییراتی در عناوین ماه‌ها و تعدیل آن‌ها در آخرین روزهای سال ۱۳۰۳ و اولین روزهای سال ۱۳۰۴ ش، نمایندگان پنجمین دوره مجلس شورای ملی گاهشماری هجری شمسی کنونی ایران را تصویب کردند. بر اساس این قانون استفاده از گاهشماری هجری شمسی با مبدأ هجرت پیامبر اسلام از مکه به مدینه و با طول سال شمسی و با نام ماه‌های فارسی میانه مورد استفاده قرار گرفت [1]. در کشور افغانستان نیز در سال ۱۳۷۷، گاهشماری هجری شمسی افغانستان، بین گاهشماری پیشین و گاهشماری هجری شمسی رایج در ایران یکسان‌سازی‌هایی صورت دادند و آرایش تعداد روزهای سال همانند گاهشماری هجری شمسی ایران شد [2].

صرفاً تغییر قرن مسئله‌ای نیست و پیش از این چنین اتفاق افتاد تا به اینکه به قرن فعلی رسیدیم. تفاوت در آن است که ما امروزه اغلب برای تمامی کارها به سیستم‌های کامپیوتری و نرم افزارها نیازمندیم. مشکل خطای هزاره تاریخ شمسی اگرچه در سخت افزار یا حتی سیستم عامل کامپیوترها مشکلی ایجاد نمی‌کند اما مشکلی بسیار جدی در نرم افزارهای مالی و اداری و هر نرم افزار دیگری است که تاریخ شمسی در آن کاربرد دارد، بوجود می‌آورد. و این مشکل ممکن است در

ابتدای اجرای یک نرم افزار باشد، دقیقاً زمانی که نرم افزار، تاریخ روز را محاسبه و نمایش دهد یک مشکل نرم افزاری پیش آمده و نرم افزار اجرا نشود.

هنوز چند سالی تا سال ۱۴۰۰ هجری شمسی فرصت باقی هست. همه ما درباره خطای هزاره سال ۲۰۰۰ میلادی تا حدودی حضور ذهن داریم، با توجه به این تجربه و نرم افزاری شدن انواع فرآیندها در سازمان ها، می توان این سوال را پرسید که آیا اول فروردین سال ۱۴۰۰ یک روز مسئله آفرین نرم افزاری خواهد بود؟ بی شک می تواند این چنین باشد و مشکلاتی رخ دهد. ولی پیشگیری ها و دقت عمل متخصصان می تواند از وقوع مشکلات جلوگیری کند، البته احتمالاً نه چندان ساده و نه کم هزینه.

در ادامه معرفی دقیق تری از مشکل خطای هزاره شمسی ارائه می شود و سپس به مسائل مرتبط و ارائه راهکار برای حل و پیشگیری می پردازیم.

۲. مروری بر پیشینه

مشکل سال ۲۰۰۰ و باگ Millennium که به مشکل هزاره معروف شده بود از اینجا نشأت گرفت که، سابقاً طراحان نرم افزار و برنامه نویسان عادت داشتند تا از یک عدد دو رقمی برای نمایش تاریخ در نرم افزارهای خود استفاده کنند [3]. مثلاً برای نمایش ۱۹۹۸ از ۹۸ استفاده می کردند. و با همین روش اطلاعات ارزشمند در کامپیوترها ذخیره شده بود. به همین علت کامپیوترها در آغاز سال ۲۰۰۰ قادر نبودند تشخیص بدهند که منظور از "00" سال 1900 است یا 2000 و اغلب با 1900 اشتباه می کردند. کیسه بودن سال 2000 [4] نیز به این مشکل افزود، چراکه برخی نرم افزارها این روز اضافه را نیز در نظر نگرفته بودند. بر اساس گزارشات هزینه برطرف کردن این مشکل و اصلاح آن (چهار رقمی کردن عدد نمایش دهنده تاریخ) بالغ بر ۳۰۰ میلیارد دلار بوده است [5]، [6]. و این به معنای رخداد یک فاجعه در دنیای نرم افزار می باشد. این مشکل باگ هزاره یا به اختصار Y2K نامیده شد و در بسیاری از نرم افزارها و حتی در برخی از نرم افزاری BIOS مادربردها وجود داشت که در شروع هزاره جدید، تاریخ به سال ۱۹۰۰ باز می گشت. این مشکل زمانی پیچیده تر شده بود که مشخص شد در بسیاری از نرم افزارها یا سخت افزارها حتی حافظه لازم برای نگهداری یا اضافه کردن دو رقم اول (یعنی قرن) وجود ندارد. قابل پیش بینی بود که چنین مشکلی می تواند باعث خطا در بسیاری از نرم افزارها و کار افتادن کامپیوترها شود. به هر حال پس از صرف میلیاردها دلار و ساعتها بررسی و اصلاحات نرم افزاری و سخت افزاری Y2K بدون مشکل جدی سپری شد.

همچنین طبق بررسی های صورت گرفته مشخص شد که سیستم های کامپیوتری ۳۲ بیتی در صورتی که تا چند سال آینده یعنی سال ۲۰۳۸ استفاده شوند نیز دچار بحران جدید بنام YK-2038 خواهند شد [7] و [8]. چراکه در این سیستم ها برای ذخیره سازی تاریخ به همراه زمان از اعداد صحیح ۳۲ بیتی استفاده می شود، که بیشترین مقدار قابل ذخیره سازی آن 2147483647 می باشد که معادل تاریخ 19-Jan-2038, 03:14:07 UTC بصورت باینری 01111111 11111111 11111111 می باشد و تنها یک ثانیه بعد از آن، زمان به تاریخ 13-Dec-1901 20:45:52 UTC که بصورت باینری 10000000 00000000 00000000 می باشد [9]. البته به نظر می رسد تا چند سال آینده از سیستم های ۳۲ بیتی استفاده نشود و حتی پس از این سیستم عامل های ۳۲ بیتی ارائه نگردند، ولی اکثر دستگاه های صنعتی و کارخانجات که مبتنی بر سیستم های کامپیوتری هستند همچنان از سیستم های ۳۲ بیتی استفاده می کنند که تغییرات در آنها هزینه های بسیار زیادی را متحمل می گردد. از آنجا که ممکن است در زمان

طراحی سیستم‌ها بعضی از جوانب از دید انسانی پنهان بماند، به نظر از این دست اشکالات همچنان به شکل‌های متفاوتی وجود خواهند داشت.

شاید تا حدودی خطای هزاره شمسی زیان‌های کمتری داشته باشد، چراکه تنها دو کشور به صورت رسمی از آن استفاده می‌کنند و دیگر اینکه سخت افزار پایه‌ای اکثر سیستم‌ها با تاریخ شمسی کار نمی‌کند. البته نباید این مسئله باعث شود ذهن ما صرفاً به نرم افزارها معطوف شود و از سخت افزار غافل شویم، چراکه سیستم‌های کامپیوتری مانند انواع دستگاه‌های تردد، سیستم‌های نوبت دهی، کنترول‌های الکترونیکی و انواع سخت افزارهای دیگر وجود دارند که اساس عملکرد سخت افزاری آن‌ها بر حسب تاریخ شمسی می‌باشد.

۳. بیان مسأله

امروزه در کشور ما بسیاری از نرم افزارها و بانکهای اطلاعاتی وجود دارند که از نوع داده‌ای (DateTime) به عنوان تاریخ یا زمان استفاده می‌کنند که مشکلی با شروع قرن جدید شمسی ندارد اما چون این نوع داده با فرمت میلادی کار می‌کنند برخی از طراحان نرم افزار ترجیح می‌دهند تاریخ شمسی را بصورت خلاصه دو رقمی ذخیره کنند. این مسأله در نرم افزارهایی که از قبل تولید شده‌اند بیشتر به چشم می‌خورد. با توجه به اینکه تاریخ شمسی در چند سال آینده با مشکلی مشابه به مشکل هزاره میلادی روبرو خواهد شد، این موضوع و تهیه راهکار آن دارای اهمیت زیادی خواهد شد. برای مثال در بسیاری از نرم افزارها هنگام دریافت، نمایش و یا هنگام ذخیره‌سازی تاریخ شمسی از شکلی مشابه 96/01/01 استفاده می‌شود. اما در سال 1400 و بعد از آن این شیوه نمایش و ذخیره تاریخ مشکل ساز خواهد شد، چراکه تاریخ به صورت 00/01/01 خواهد بود که سیستم کامپیوتر نمی‌داند منظور از "00" به سال 1400 مربوط می‌شود یا سال 1300 و در صورتی که چنین پیش برویم با سال 1300 اشتباه خواهند گرفت. این مسئله یک مشکل اساسی دیگر خواهد داشت و آن کبیسه نبودن سال 1400 است در صورتی که سال 1300 کبیسه خواهد بود [10]. چراکه نرم افزارها یک روز اضافه برای آن سال در نظر خواهند گرفت در صورتی که اینطور نیست. اگر از همین امروز سعی شود چنین تاریخی که در بسیاری از نرم افزارها بصورت ۸ رقمی دریافت می‌کنند وارد شود به احتمال زیاد با پیام خطا و عدم صحت تاریخ مواجه خواهد شد و یا در زمان عملیات گزارشگیری با مشکلاتی مواجه خواهد شد و خروجی نادرستی را ایجاد می‌کند.

جدول ۱- تولید رکوردهای تکراری

ب- رکورد مربوط به قرن ۱۴ هجری شمسی

| سال | ماه | روز |
|-----|-----|-----|
| 60 | 01 | 01 |

الف- رکورد مربوط به قرن ۱۳ هجری شمسی

| سال | ماه | روز |
|-----|-----|-----|
| 60 | 01 | 01 |

در حالت کلی انواع نقص‌هایی که در سیستم نرم‌افزاری دارای خطای هزاره شمسی ممکن است بوجود بیایند به صورت زیر می‌باشند (شکل ۱):

۱- رکوردهای تکراری

مشکل ثبت رکوردهای تکراری در یک تاریخ زمانی رخ می‌دهد که فقط سال مربوط به تاریخ شمسی ذخیره شود و قرن نادیده گرفته شود، که منجر بوجود آمدن چند رکورد مشابه خواهد شد. برای مثال رکوردی که در سال ۱۳۶۰/۰۱/۰۱

(جدول ۱- الف) و رکوردی که در سال ۱۴۶۰/۰۱/۰۱ (جدول ۱- ب) یکسان خواهند بود. چراکه قرن به همراه سال ذخیره نشده است و هر دو رکورد با مقدار یکسان ۶۰/۰۱/۰۱ ذخیره خواهند شد که باعث تولید رکوردهای تکراری در یک تاریخ خواهد شد.



شکل ۱- انواع نقص در سیستم‌های دارای خطای هزاره شمسی

۲- ثبت نشدن رکوردها

ممکن است در سیستم نرم افزاری به دلیل تکراری فرض شدن، رکورد ثبت نگردد. همان مثال قبلی را در نظر بگیرید با این تفاوت که ممکن است رکوردی ثبت نگردد. چراکه ممکن است در سیستم نرم افزاری قبل از ثبت رکورد جدید بررسی شود که آیا در یک تاریخ، رکوردی از قبل وجود دارد و در صورت وجود، رکورد ثبت نگردد. این امر باعث می‌شود رکوردی که برای قرن ۱۴ هست با رکوردهای از قبل موجود که برای قرن ۱۳ هست اشتباه گرفته شود.

۳- اشتباه در گزارشگیری

برای عدم پاسخ دهی مناسب گزارشگیری‌ها فرض کنید به رکوردهای یک جدول از پایگاه داده‌ها از سال 1396 الی سال 1401 نیاز داریم و با در نظر گرفتن این نکته که سیستم نرم افزاری دارای خطای هزاره شمسی است و سال بصورت دو رقمی "96" الی "01" ذخیره شده است که SQL Query آن بصورت زیر می‌شود.

SELECT * From Table1 Where (Year >= 96) AND (Year <= 01) (۱)

این پرس و جو هیچ خروجی‌ای را بر نمی‌گرداند، چراکه هیچ رکوردی یافت نمی‌شود که سال آن بیشتر از ۹۶ و کوچکتر از ۱ باشد. البته این گزارش ممکن است برای رکوردهای مربوط به قرن ۱۳ شمسی را به درستی پاسخ دهد.

SELECT * From Table1 Where (Year >= 90) AND (Year <= 96) (۲)

این پرس و جو به درستی رکوردهایی که سال آن‌ها بین ۹۰ الی ۹۶ (مربوط به قرن ۱۳) است را نشان می‌دهد. البته اگر در این بین رکوردهایی از قرن ۱۴ وجود نداشته باشند، که اگر چنین باشد نیز مشکلی دیگر رخ داده است.

۴- فیلدهای متغیر

ثابت مقادیر فیلدها با طول متغیر نیز از دیگر مشکلات ثبت سال شمسی دو رقمی خواهد بود. فرض کنید در سیستمی دیگر، عملیاتی بر اساس تاریخ دو رقم سال صورت پذیرد. برای مثال یک عدد ۱۱ رقمی مانند 83012345678 در سیستم آموزشی دانشگاه که بر اساس سال و ترم ثبت نام به دانشجویان شماره دانشجویی اختصاص می‌دهد، از چند حالت زیر خارج نیست:

الف- فیلدی از نوع عدد صحیح برای نگهداری عدد در نظر گرفته شده باشد از آنجاکه برای اختصاص شماره دانشجویی به دانشجویان دو صفر سمت راست مبنا قرار داده می‌شود. و در سال 1400 نیز دو رقم 00 قبل از عدد بی معنا خواهد بود، عدد 00012345678 بصورت عدد 12345678 ذخیره خواهد شد که طول این عدد ۸ رقم خواهد شد. این مسأله ایجاد اعداد با طول متغیر و حتی پس از گذشت چند سال باعث رخ دادن مشکل رکوردهای تکراری خواهد شد. در مورد طول متغیر باید گفت ممکن است در نرم افزار یک بررسی کننده طول عدد ورودی نیز وجود داشته باشد که نیاز به اصلاحات داشته باشد که محدودیتی در ورودی طول اعداد ورودی نداشته باشد.

ب- فیلدی از نوع کاراکتری برای نگهداری عدد در نظر گرفته شده باشد که در اینصورت مشکل ایجاد اعداد با طول متغیر وجود ندارد ولی مشکل بوجود آمدن رکوردهای تکراری همچنان باقی هست.

۵- خطای منطقی

این مشکل زمانی در سیستم نرم افزاری دارای خطای هزاره شمسی رخ می‌دهد که صرفاً درون برنامه عملیات‌ها یا بررسی‌ها و یا تصمیم گیری‌ها مطابق سال شمسی دو رقمی صورت پذیرد. برای مثال یک دستور if که صرفاً براساس سال دو رقمی کار می‌کند و بصورت زیر است را در نظر بگیرید.

```
int CurrentYear = 99; //1396
int NextYear = 0; //1400
Input (x);
if ( x > CurrentYear && x < NextYear ) (۳)
{
    //Statements
}
```

در این مثال می‌بینیم که اگر x برابر با ۹۷ باشد دستورات قسمت Statements مربوط به if اجرا نخواهند شد، چراکه هیچ مقداری برای x نمی‌توان یافت که از 96 بزرگتر و از 0 کوچکتر باشد. در نظر داشته باشید در نوع داده‌ای عددی، دو رقم 0 نیز بصورت یک رقم 0 ذخیره خواهد شد.

۶- خطای محاسباتی

این خطا زمانی رخ می‌دهد که در کدهای برنامه نویسی بر اساس سال شمسی دو رقمی محاسباتی صورت پذیرد. برای مثال بخواهیم براساس سال فعلی و سال شروع ثبت رکورد، تعداد سال‌های ثبت رکورد را بدست آوریم.

```
int StartYear = 96;           //1396
int CurrentYear = 0;         //1400
int Count = CurrentYear – StartYear; //-96
```

(۴)

می‌بینیم که خطایی رخ داده است که ما انتظار آن را نداشتیم، حتی اگر قدر مطلق آن محاسبه شود باز هم عدد ۹۶ بدست می‌آید که جواب مد نظر ما نیست. تمامی این مشکلات صرفاً در اثر در نظر گرفتن مقدار سال شمسی بصورت دو رقمی رخ می‌دهند.

از آنجایی که در ایران چرخه استفاده از نرم افزار طولانی بوده و پشتیبانی نیز به نسبت ضعیف است این احتمال وجود دارد که در سال ۱۴۰۰ نرم افزارهای زیادی وجود داشته باشند که با مشکل تاریخ مواجه شده و به تیم پشتیبانی و طراحان نرم افزار نیز دسترسی وجود نداشته باشد، بنابراین لازم است با چنین چشم اندازی از همین امروز در تولید، انتخاب، توسعه و خرید نرم افزار این مسئله در نظر گرفته شود. اهمیت این موضوع بسیار زیاد است و لازم است هر چه زودتر در سیستم‌های سخت افزاری کامپیوتری، نرم افزارهای کاربردی، وب سایت‌ها و انواع نرم افزار که سال شمسی را بصورت ۲ رقمی ذخیره می‌کنند این مشکل بررسی و اصلاحات لازم صورت گیرد. بایستی به این سمت حرکت کنیم که نمایش و دریافت تاریخ بصورت ۱۰ رقم و با قالب YYYY/MM/DD انجام گیرد و سال چهار رقمی بجای دو سال رقمی دریافت شود. همچنین بانکهای اطلاعاتی بررسی و فیلهای تاریخ اصلاح شده و تاریخ بصورت ۱۰ رقمی ذخیره شود. البته مشکل خطا هزاره شمسی در همه نرم افزارهای کامپیوتری بومی وجود ندارد و نرم افزارهایی که استاندارد ذخیره سازی تاریخ را رعایت کرده باشند از این امر مستثنی هستند. برای مثال، در بسیاری از نرم افزارها از تاریخ شمسی صرفاً برای نمایش و راهنمای کاربران در نمایش و گزارشات استفاده می‌شود و در عمل و پشت پرده اساس انواع عملیات‌ها بر اساس نوع داده‌ای تاریخ و ساعت (DateTime) مربوط به زبان‌های برنامه نویسی و بانکهای اطلاعاتی استفاده می‌گردد که مشکلی با خطای هزاره ندارند. جدول ۲ نمونه‌ای از داده‌های ذخیره شده چنین سیستم‌های نرم افزاری را نشان می‌دهد.

جدول ۲- رکوردهای ثبت شده سیستم نرم افزاری فاقد خطای هزاره شمسی

| ID | Title | Description | Year | Month | Day | MyDateTime |
|-----|--------|-------------------------|------|-------|-----|------------------------|
| 1 | Title1 | Description of record 1 | 95 | 08 | 03 | 10-24-2016 10:53:12 AM |
| 2 | Title2 | Description of record 2 | 96 | 09 | 03 | 11-24-2017 12:50:11 AM |
| 3 | Title3 | Description of record 3 | 96 | 09 | 04 | 11-25-2017 12:21:57 PM |
| 4 | Title4 | Description of record 4 | 96 | 12 | 28 | 19-03-2018 08:32:53 PM |
| 5 | Title5 | Description of record 5 | 00 | 01 | 01 | 21-03-2021 06:50:07 AM |
| 6 | Title6 | Description of record 6 | 00 | 02 | 01 | 21-04-2021 09:01:11 AM |
| 7 | Title7 | Description of record 7 | 00 | 12 | 29 | 20-03-2022 11:10:23 PM |
| 8 | Title8 | Description of record 8 | 01 | 01 | 03 | 23-03-2022 10:23:20 PM |
| ... | ... | ... | ... | ... | ... | ... |

همانطور که مشاهده می‌شود در این جدول جدای از ذخیره سال شمسی به صورت ۲ رقمی، تاریخ بصورت میلادی DateTime ذخیره می‌شود و مبنای تمامی عملیات‌های نرم افزار قرار دارد و از تاریخ شمسی در این سیستم صرفاً برای نمایش و راهنمایی کاربر استفاده می‌گردد. چنین سیستم‌هایی با اینکه سال شمسی را به صورت ۲ رقمی ذخیره می‌کنند

دارای خطای هزاره شمسی نخواهند شد. در صورتی که همین سیستم تاریخ بصورت DateTime را نداشته باشد، در چند سال آینده دچار خطای هزاره شمسی خواهد شد.

۴. راه حل و فرآیند پیشنهادی

چند راه برای حل مشکل خطای هزاره در سطح پایگاه داده وجود دارد که به صورت موارد زیر شرح داده می‌شود:

۱- تبدیل مقادیر تمامی سال‌های ۲ کاراکتری به سال ۴ کاراکتری در صورتی که فیلد داده‌ای مربوط به سال بصورت کاراکتری باشد و سپس الحاق مقادیر موجود با کاراکترهای قرن "13" که SQL Query آن بصورت زیر می‌باشد.

UPDATE Table1 SET Year = '13' + Year (۵)

۲- افزودن مقدار (۱۰۰× قرن) به تمامی مقادیر سال‌ها در صورتی که فیلد داده‌ای مربوط به سال بصورت عددی باشد کافی است مقدار قبلی آن برای مثال با مقدار ۱۳۰۰ جمع شود. فرض کنید اگر مقدار قبلی آن ۹۶ باشد با ۱۳۰۰ جمع شود نتیجه حاصل برابر ۱۳۹۶ خواهد شد. که SQL Query آن بصورت زیر می‌باشد.

UPDATE Table1 SET Year = 1300 + Year (۶)

۳- افزودن فیلد جدید بنام قرن و مقدار دهی تمامی رکوردهای آن با "13"

ALTER Table Table1 Add Century int ; (۷)

UPDATE Table1 SET Century = 13;

همچنین برای حل مشکل محاسباتی خطای هزاره باید کدهای برنامه نویسی مربوط به نرم افزار یا سیستم نرم افزاری اصلاح شود. و در صورتی که مبنای محاسباتی بصورت تاریخ شمسی دو رقمی است، به صورت چهار رقمی تغییر یابد، یا می‌توان سال را بصورت سه رقمی در نظر گرفت که حداقل تا چند قرن دیگر مشکلی نخواهد داشت.

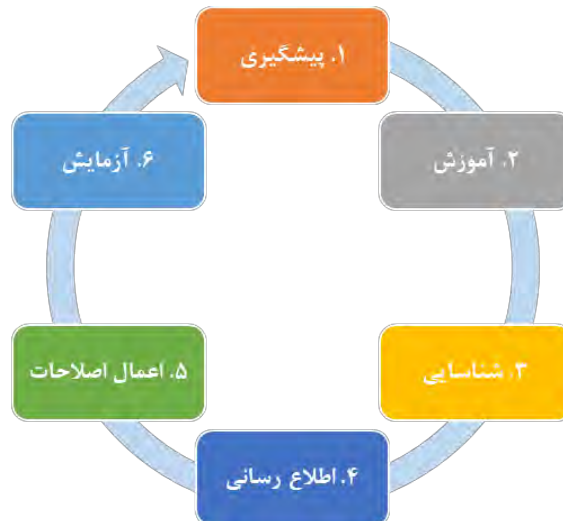
نبود شناسنامه نرم افزاری و عدم وجود پشتیبانی نرم افزاری مناسب از جمله مشکلاتی هستند که ارائه راهکار برای حل مشکل در زمانی مناسب و با هزینه‌ای کم را تحت الشعاع قرار می‌دهند. به همین دلیل برای رفع و گذر از مشکل خطای هزاره نیاز به یک فرآیند منظم خواهیم داشت. فرآیند پیشنهادی مورد نظر در ادامه و بصورت زیر ارائه می‌گردد (شکل ۲).

۱. پیشگیری از بوجود آمدن سیستم‌های جدید با خطای هزاره
اولین مورد در مقابله با خطای هزاره شمسی این است که از بروز خطاهای نرم افزاری جدید پیشگیری کنیم برای این کار باید:

- ۱.۱. آگاهی به تمامی مسئولین فناوری اطلاعات سازمان ها از مشکل خطای هزاره
- ۱.۲. ابلاغ به تنظیم بندی در قراردادهای جدید سیستم‌های کامپیوتری مبنی بر خطای هزاره شمسی

۲. آموزش مسئولین فناوری اطلاعات سازمان‌ها

شاید بزرگترین مشکلی اکثر سازمان‌های ما با آن مواجه هستند، مسأله پیدا کردن برنامه‌های دارای خطای هزاره موجود و به کاربرده شده در سازمان است. برای حل این مشکل باید گروهی جهت آموزش به مسئولین نرم افزارها، آماده شود تا آن‌ها بتوانند آموزش‌های لازم جهت انجام مراحل بعدی را بدهند.



شکل ۲- فرآیند پیشنهادی

۳. شناسایی و یافتن سیستم‌های دارای خطا هزاره در سازمان‌ها
 - ۳,۱,۱. مشاهده قراردادهای و مستندات نرم افزارها و سیستم‌های کامپیوتری
 - ۳,۱,۲. بررسی ظاهری فرم‌های ورود اطلاعات نرم افزارها و سیستم‌های نرم افزاری در دریافت تاریخ‌های ورودی، که آیا سال ورودی با ۲ عدد دریافت می‌شود یا به صورت ۴ عدد
 - ۳,۱,۳. بررسی کدهای منبع نرم افزارها در صورت وجود
 - ۳,۱,۴. بررسی ساختاری و داده‌های پایگاه داده سیستم‌های نرم‌افزاری (در صورت دسترسی)
 - ۳,۱,۵. ستعلام از شرکت یا فرد سازنده سیستم نرم افزاری (در صورت دسترسی)
 - ۳,۱,۶. بررسی امکانپذیر بودن اصلاحات (در صورت عدم اصلاح پذیری، بررسی ساخت مجدد سیستم نرم افزاری یا نرم افزار مورد نظر)
 - ۳,۱,۷. امکان سنجی زمانی و هزینه‌ای اصلاحات
 - ۳,۱,۸. ارائه راهکار جهت جایگزینی موقت خدمات
 - ۳,۲. تهیه لیستی از سازمان‌های دارای مشکل خطای هزاره به همراه نوع خدمات نرم افزار یا سیستم نرم افزاری، زمان و هزینه‌های پیش بینی شده اصلاحات
۴. اطلاع رسانی به استفاده کنندگان خدمات مبنی بر خارج از دسترس بودن خدمات در بازه زمانی مشخص شده جهت انجام اصلاحات و معرفی راهکار جایگزین

به هر حال برای انجام اصلاحات یا جایگزین کردن خدمات مدت زمانی لازم است، و ممکن است افرادی در حال استفاده از آن خدمات باشند و عملیات‌های متفاوتی را با آن سیستم نرم افزاری انجام دهند. از دسترس خارج کردن

خدمات این نرم افزار یا سیستم نرم افزاری مشکلاتی را بوجود خواهد آورد. برای کاهش شدت این مشکلات باید اطلاع رسانی‌های به افراد مرتبط با آن سیستم از قبل صورت گیرد و در صورت نیاز راهکار جایگزین معرفی گردد.

۵. اعمال اصلاحات نرم افزاری یا جایگزینی سیستم‌های نرم افزاری

در این زمان باید اصلاحاتی که از قبل مشخص شد صورت گیرد. در نظر داشته باشید که گاهی انجام این اصلاحات برای سیستم مقدور نیستند. برای مثال فرض کنید سیستم قابلیت اصلاح نداشته باشد یا کدهای برنامه نویسی آن به راحتی موجود نباشد و یا برای دسترسی به کدهای آن هزینه زیادی درخواست شده باشد. در چنین مواقعی جایگزینی سیستم نرم افزاری دیگر و یا ساخت مجدد آن می‌تواند مقرون به صرفه‌تر باشد.

۶. آزمایش برنامه‌های درون سازمانی و برون سازمانی

آخرین مرحله مربوط به شروع تست نهایی برنامه‌هایی می‌باشد که داخل سازمان تهیه شده و هم چنین از بیرون خریداری شده است می‌باشد. که آیا اصلاحات صورت گرفته برای نرم افزار یا سیستم نرم افزاری مورد نظر به درستی با مشکل خطای هزاره شمسی برخورد می‌کنند.

۵. نتیجه‌گیری

مشکل خطای هزاره شمسی یک مشکل انکار ناپذیر خواهد بود. برای روبرویی با چنین مشکلاتی باید پیش بینی‌هایی انجام دهیم و خود را برای مقابله با آن آماده سازیم. چنین امری نیازمند هماهنگی تمامی دستگاه‌های نظارتی و دولتی را می‌طلبد. و به نظر بدون هماهنگی و آموزش درست، مقابله با این مشکل به درستی و در زمان مناسب شدنی نباشد. ما در این مقاله به بیان دقیق مشکل فوق پرداختیم، سپس راه‌حل‌هایی برای آن ارائه دادیم و در نهایت فرآیندی مبتنی بر همکاری سازمان‌ها و دستگاه‌های نظارتی را پیشنهاد کردیم.

۷. مراجع

۱. عبدالمهدی، ر. (۱۳۶۶)، "تاریخ تاریخ در ایران"، تهران، صفحه ۳۳۱.
2. Tsybulsky, V. (1979), "Calendars of Middle East countries: conversion tables and explanatory notes," Moscow, p 233.
3. Schumacher, M. (1999), "Year 2000, Y2K, millennium bug," Power Engineering Society Summer Meeting, Erlangen, Germany.
4. "Gregorian Calendar," (17 December 2017). [Online]. Available: <https://msdn.microsoft.com/en-us/library/system.globalization.gregoriancalendar.aspx>.
5. Mitchell, R. L. (2009), "Y2K: The good, the bad and the crazy," ComputerWorld (December 2009).
6. Project Federal Reserve Bank of Minneapolis Community Development. (2017), "Consumer Price Index (estimate) 1800," Federal Reserve Bank of Minneapolis.

7. Sharma, V. C. (2015), "YK-2038 Bug," Journal Of Image Processing And Artificial Intelligence, vol. 1, no. 3.
8. Bergmann, A. (2015), "The End Of Time (32bit edition)," in Embedded Linux Conference, San Jose.
9. Harshini , S. Kavyasri, K.R. Bhavishya, P. Sethukkarasi, T. (2017), "Digital World Bug: Y2k38 an Integer Overflow Threat-Epoch," International Journal of Computer Sciences and Engineering, vol. 5, no. 3.

۱۰. صیاد، م. ر. (۱۳۹۲)، "ترتیب و توالی سال‌های عادی و کبیسه در بازه زمانی ۳۹۲۰ سال هجری شمسی (۹۴۰ - تا ۲۹۷۹)،" میراث علمی اسلام و ایران، جلد سال دوم، شماره اول (پیاپی ۳)، بهار و تابستان ۱۳۹۲.