سر و کله زدن با ارقام بزرگ

بازی بزرگان

Pix << p 13 - prog-main >>
عکس: purdue.edu

امیربهاالدین سبط‌الشیخ

یکی از مسایلی که بیشتر برنامه‌نویسان، دیر یا زود با آن روبه‌رو می‌شوند، ذخیره‌سازی و محاسبه اعداد و ارقام بزرگ است. همان‌طور که می‌دانیم، ظرف ذخیره‌سازی داده‌ها متناسب با حجم و میزان بزرگی ارزش داده، رشد می‌کند.

به‌طور مثال، در زبان C بزرگ‌ترین حافظه‌ای که برای ذخیره‌سازی داده‌ها استفاده می‌شود، برابر با 8 بایت است که این مقدار در کامپایلر C ANCI به 10 بایت افزایش پیدا کرده است. اما زمانی پیش می‌آید که نیاز به ذخیره‌سازی و محاسبه اعداد بزرگ‌تر از 8 بایت داریم.

یکی از ساده‌ترین روش‌ها، ذخیره‌سازی ارقام اعداد در آرایه و انجام عملیات مورد نظر روی آرایه‌هاست. حالا بیایید ببینیم که چگونه ۲عدد بزرگ را که در آرایه ذخیره شده‌اند با هم جمع کنیم.

ما اعداد را به‌صورت آرایه‌ای از کاراکترها از کاربر دریافت می‌کنیم. برای کم شدن فضای اشغال شده از کاراکتر استفاده می‌کنیم و با یک تبدیل ساده، خانه‌های آن را به رقم مورد نظر تبدیل می‌کنیم.

اعداد را به این صورت از کاربر دریافت می‌کنیم:

```
char num1[1000],num2[1000];
 std::cout << "Enter First Number : " << std::endl;
 std::cin >> num1;
 std::cout << "Enter Second Number : " << std::endl;
 std::cin >> num2;
```

به این نکته توجه داشته‌باشید که این جا اعداد ما 1000 رقمی هستند و میزان حافظه اشغالی برای آن‌ها با کوچک‌تر شدن‌شان تغییری نمی‌کند، ما می‌توانیم با تعریف اعداد به‌صورت اشاره‌گر برای آن‌ها حافظه داینامیک در نظر بگیریم و هر عدد بتواند هر تعداد رقمی را که خواست داشته‌باشد.

حالا باید ۲عدد را جمع کنیم، این کار را تابعی با نام add انجام می‌دهد که به این‌صورت تعریف شده است:

```
void add(char num1[],char num2[]){
   int len =
getMax(getLength(num1),getLength(num2));
   reverse(num1);
   reverse(num2);
   fillzero(num1,len-getLength(num1));
   fillzero(num2,len-getLength(num2));
   int index = 0;
   int carry = 0;
   for(index ; index < len ; index++){
      int res = (num1[index] - 48) + (num2[index] -
48) + carry;
      if(res > 9){
         carry = res / 10;
         res = res % 10;  }
      else{
         carry = 0;  }
      num1[index] = (res + 48);  }
   num1[index + 1] = '\0';
   reverse(num1);
}
```

این تابع ابتدا طول عدد بزرگ‌تر را پیدا می‌کند، سپس اعداد را بر عکس می‌کند؛ اما چرا برعکس؟

کاربر عدد 12 را وارد کرده است. این عدد به این‌صورت در آرایه قرار گرفته است:

			2	1

۴خانه باقی‌مانده بلااستفاده هستند. ما توسط تابع reverse خانه‌های آرایه را جابه‌جا می‌کنیم و سپس به اندازه اختلاف طول ۲عدد با

طول بیشتر یا همان len در کد بالا صفر اضافه می‌کنیم. این‌کار را برای این انجام می‌دهیم که اگر طول ۲عدد متفاوت بود، بتوان ۲عدد را با هم جمع کرد (البته بدون اضافه کردن صفر هم می‌توانیم این کار را انجام دهیم، ولی این‌کار جمع کردن را ساده‌تر می‌کند). بعد از این مرحله، در حلقه for تک‌تک ارقام را با هم جمع می‌کنیم. مقدار متغییر carry مقدار رقم نقلی از جمع ۲عدد حاصل است.

مثلا اگر 2 با 9 جمع شود، مقدار carry برابر یک است که مقدار آن در جمع ۲رقم بعدی استفاده شود. در نهایت، حاصل جمع ۲عدد در آرایه num1 قرار می‌گیرد و سپس باید num1 نیز بر عکس شود تا حاصل، نتیجه دلخواه ما باشد. این روش، تنها یکی از روش‌های جمع ۲ عدد بزرگ است و مفهوم جمع کاغذی را پیاده می‌کند.

بدیهی است روش‌های بهینه‌تر و با مصرف کمتر حافظه وجود دارند که در صورت امکان، در آینده آن‌ها را توضیح خواهیم داد. کد این برنامه نیز از طریق لینک زیر قابل دریافت است:

http://tinyurl.com/clickprog261

پرسش و پاسخ

نداده است.

به‌نظر می‌آید این آنتی‌ویروس‌ها ویروس‌کشی نکرده‌اند، زیرا اگر چنین کاری می‌کردند مشکل شما حل شده بود. اما برای حل مشکل، باید دید سیستم شما دچار چه نوع ویروس، بدافزار یا Trojan شده است و سپس باید برای رفع مشکل اقدام کرد. آنتی ویروس‌هایی که استفاده کرده‌اید، اگر به‌روز باشند، جزو بهترین آنتی‌ویروس‌ها هستند. ممکن است سیستم شما مشکل دیگری غیر از ویروس هم داشته باشد. اگر ویروس‌ها را شناسایی کرده‌اید، نام آن‌ها را عنوان کنید تا برای هرکدام راه‌حلی ارایه شود.

امید پاکدل از اهواز - سیستم کامپیوتر من روشن می‌شود، یک سری بوق سریع می‌زند و سیستم بالا نمی‌آید. مشکل چیست؟

هنگامی که یک کامپیوتر روشن می‌شود و برای بالا آمدن آماده می‌شود یا زمانی که کامپیوتر دوباره راه‌اندازی می‌شود، BIOS سیستم با عملکردی به نام POST به تست سخت‌افزار سیستم می‌پردازد تا از درستی اتصال انواع Deviceها مطلع شود.

بنابراین اگر سیستم با مشکلی مواجه شود، پست برای BIOS پیغام خطا می‌فرستد. این پیغام‌ها همان بوق‌ها هستند که باعث اطلاع شما از بروز مشکل می‌شوند. بایوس سیستم‌های مختلف از سوی شرکت‌های مختلفی تولید می‌شوند. در زیر بوق‌های مختلف در ۲ نوع مادربورد ارایه می‌شوند:

کدهای بوق مادربورد نوع AMI

۱ بوق: اشکال از DRAM است، بنابراین مشکلی در حافظه سیستم به‌وجود آمده است.

۲ بوق: حافظه تعادل ندارد. تساوی گردش در حافظه به‌درستی کار نمی‌کند.

۳ بوق: ۶۴ کیلوبایت مورد نیاز برای راه‌اندازی سیستم در دسترس نیست که به رم سیستم برمی‌گردد.

۴ بوق: تایمر سیستم شما با مشکل مواجه شده است که مانع از درستی همزمان عملکرد مادربرد می‌شود.

۵ بوق: پردازنده مشکل دارد.

۶ بوق: Controller ورودی A20/Keyboard با مشکل دارد. مدار کنترل‌کننده صفحه کلید با مشکل مواجه شده است.

۷ بوق: حافظه مجازی با مشکل مواجه است.

۸ بوق: حافظه کارت ویدئو مشکل دارد. بایوس نمی‌تواند روی فریم‌های بافر کارت ویدئو بنویسد.

۹ بوق: ROM شما مشکل دارد، چیپ رام‌بایوس روی مادربرد دچار خطا است.

۱۰ بوق: CMOS مشکل دارد. ایرادی در مادربرد مانع از فعل و انفعال با سی‌ماس می‌شود.

۱۱ بوق: Cache در Level 2 حافظه دچار مشکل شده است.

یک بوق بلند و ۲ بوق کوتاه: سیستم ویدئویی دچار مشکل شده است.

یک بوق بلند، ۳ بوق کوتاه: ایرادی مانند مشکل در ۳ بوق دارد.

یک بوق بلند، ۸ بوق کوتاه: عملگر تست‌کننده خطا دارد.

بوق مداوم: اشکال از رم یا از ویدئو است.

کدهای بوق مادربورد نوع Phoenix

این شرکت از بوق‌های دنباله‌دار استفاده می‌کند. برای مشخص کردن فاصله بوق‌ها از خط تیره - استفاده کرده‌ایم. به معنای یک توقف و ادامه. ۱-۲-۳ یعنی اول یک بوق، سپس ۲ بوق و نهایتا هم ۳ بوق. ورژن چهارم به پایین این محصول از ۳ بوق و از ورژن ۴ به بالا از ۴ بوق استفاده می‌کند.

۱-۱-۳-۱ یا ۱-۱-۲-۱ مادربرد یا پردازنده مشکل دارد.

۱-۱-۲-۳ یا ۱-۱-۳-۱ یا ۱-۱-۳-۲ یا ۱-۱-۳-۳: مادربرد یا یکی از اجزای داخلی مشکل دارد.

۱-۱-۳-۲ یا ۱-۱-۳-۳ یا ۱-۱-۳-۴: اشکال از حافظه ۶۴ کیلوبایتی است.

۱-۱-۴-۱: Cache در Level 2 حافظه دچار مشکل شده است.

۱-۱-۴-۳: اشکال از پورت I/O است.

۱-۲-۱-۱: مشکل از Power Management است.

۱-۲-۱-۲: یا ۱-۲-۱-۳: یکی از اجزای داخلی مشکل دارد.

۱-۲-۲-۱ یا ۱-۲-۲-۳: رام‌بایوس یا ورودی صفحه کلید مشکل دارد.

۱-۲-۳-۱: تایمر سیستم ایراد دارد.

۱-۲-۳-۳: مشکل از DMA است.

۱-۲-۴-۱: مشکل از IRQ است.