

月刊

本誌正式名称は「シェルスクリプトマガジン vol.22」です
Shell Script Is The Supreme Glue Technology
FOR THE SOPHISTICATED SHELL SCRIPTERS
世界で唯一の

シェルスクリプトマガジン



2015
2 January Vol. 22
¥500

特集

探訪記!



北のDC

USP友の会出張一撃勉強会レポート



法林浩之の
FIGHTING TALKS

ユニケージ開発手法
コードレビュー

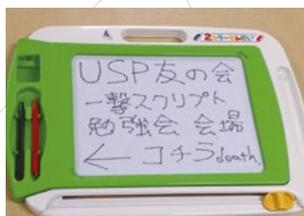
最終回!
シェルスクリプト大喜利

新連載

UNIX大好き人間はどう育つ?

スズラボ通信

CONTENTS



世界で唯一の

月刊シェル
スクリプト
マガジン

2015 February vol.22

- 04 **特集 北のDC探訪記!&USP友の会出張一撃勉強会レポ**
濱田康貴
- 14 **うにつくすなやつら 第7回**
長谷川猛
- 18 **新連載 UNIX大好き人間はどう育つ?**
鳥居高之
- 20 **ユニケージ開発手法 コードレビュー 第11回**
大内智明
- 25 **中小企業手作りIT化奮戦記 第15回**
菅雄一
- 30 **人間とコンピュータの可能性 第22回**
大岩元
- 32 Haskell版Open usp Tukubai完成させるぞ企画
Haskellでやってはいかんのか? 第11回
上田隆一
- 36 組織文化を変える汗と涙の物語
アジャイル改善塾 第10回
山海一剛
- 38 **スズラボ通信 第14回**
すずきひろのぶ
- 42 **IT美女図鑑-鎌田あすか**
- 44 未来に生きる!現場で使える!
データモデリング 第12回
熊野憲辰
- 47 **姐のBENTO**
- 48 **漢のUNIX 第18回**
後藤大地
- 56 **法林浩之のFIGHTING TALKS 第9回**
法林浩之
- 58 **シェルスクリプト大喜利 第15回**
- 62 **長続きの秘訣**
シェル魔人/編集後記

本誌名称について

vol.21より「USPMAGAZINE」から「シェルスクリプトマガジン」に変更となりました。バックナンバーは「USP MAGAZINE」としてお求めいただけます。

本誌正式名称は「シェルスクリプトマガジンvol.22」となります。



ちんじゅうちゃん

二月号の特集では、
北海道でのデータセンター見学と二撃勉強会をレポート。

雪消月

北の大地でレッツサーバー構築！



北のDC探訪記!

USP友の会出張一撃勉強会レポート



Writer : 濱田康貴

みなさんこんにちは。@nullpopopo でございます。USP友の会をよくご存知の方は、友の会=シェル芸というイメージが強く、友の会という名前を知らなくても「シェル芸」「危険シェル芸」というアブノーマルワードをTwitter上で見かけることが多いかと思いますが、「そろそろシェル芸以外の勉強会をやりたいよね。」という声があがり、2014年9月から「一撃シェルスクリプト勉強会」をはじめました。その際、さくらインターネットさんには、会場を貸していただいたり、開発環境としてさくらのクラウドを利用させていただいており、一撃スクリプトが動くクラウドのサーバーがどのようなものかを知ることでサーバー構築の理解が深まるであろうということで、この度、石狩データセンター（以下石狩DC）の見学をさせていただけることになりました。また、合わせて第2回一撃サーバー構築シェルスクリプト勉強会（さっぽろ!）も札幌で開催して来ましたので、こちらの様子もレポートします。



北海道を訪れた11月初旬は紅葉が綺麗でしたが、石狩DC訪問の日は起きたらなーんかしべれるなーと思ったら朝から雪がチラついておりました。わやですね。



地下鉄の駅からタクシーで石狩DCへ向かったのですが、ちょっと離れたところから撮影しても建物が横に広がってすべてが収まりきりません。

この日は宮下センター長と広報の室野さんにご案内いただきました。



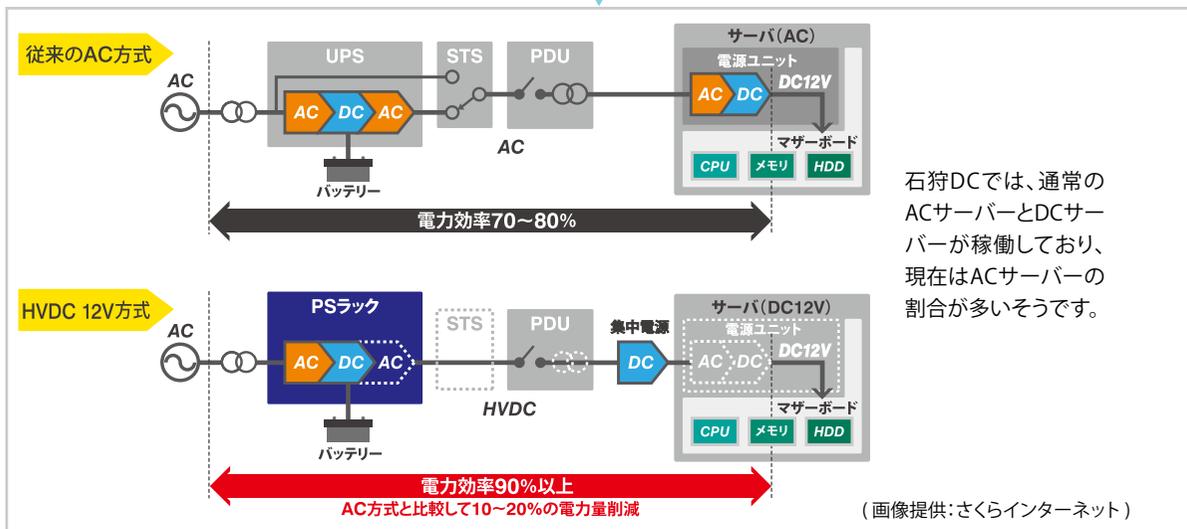
エントランスを入ると、このようなオシャレなゲートを通過します。サーバーやネットワーク機器などの搬入は別の搬入口から行うそうで、後ほど触れますが、ここにも一つ工夫があるそうです。



ここからAC6600Vに降圧され、高圧トランスでさらにAC400Vへ降圧します。ACサーバーではUPSへAC400Vで入力し、ACからDCへ変換してバッテリーへ蓄電し、さらにDCからACへ変換しAC230Vを出力します。そこからさらにサーバー電源ユニットでは230Vを入力し、DC12Vを出力します。一方DCサーバーではPSラックへAC400Vで入力し、ACからDCへ変換してバッテリーへ蓄電し、DC340Vをサーバーラック内にある集中電源ユニットへ送り出します。集中電源ユニットでは、DC340VからDC12Vへ変換し、サーバーへ送り出します。



石狩DCの電源は66000Vの特別高圧で受電しています



空調は常に20度に保たれており、搬入された機器はここに1日半程度寝かせてからサーバーラックに搬入します。

さて、先ほど機器搬入について工夫があると書きましたが、こちらが機器の搬入口となります。



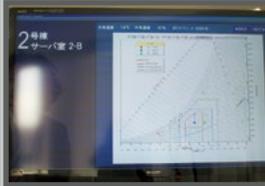
冬場は屋外と屋内の気温差が激しく、結露を防止するのが目的だそうです。また、搬入直後の機器を段ボールから出しても素手で持てないくらい冷たいのも、ここで寝かせておく理由の1つだそうです。北海道ならではの工夫ですね。



宮下センター長と私が立っている先にあるのがターボ冷凍機です。通常は外気を取り込んでサーバー室を冷却するのですが、夏場の外気が高温多湿になった際にはこれを動かして冷気を取り込みます。2011年の開所時には年間100時間の稼働を想定していたのですが、2014年の夏は1500時間を超えた稼働だったそうです。

北のDC探訪記!

USP友の会出張一撃勉強会レポ



監視センターにはPUE(※1)や外気温がプロットされるモニターが表示されているのですが、1号棟がフルに稼働し、2号棟にもラックが入り始めている状態でPUEが1.08~1.26でした。



石狩DCの外気冷房は壁吹き出し方式と天井吹き出し方式の2種類で、取り入れた外気をサーバールームの壁から真横に送り出すか、天井から送り出すかの違いがあります。

冷却効率はどちらもそれほど変わらないそうですが、壁吹き出し方式は大きなファンの少数設置でよい反面、作業者が寒い、紙のマニュアルが飛ばされるといったデメリットがあるそうです。天井吹き出し方式は壁吹き出し方式のデメリットを解消できるものの、小さなファンやラック上部の排気ダクトをラック列ごとに取り付けなければならないそうです。



外気を取り込む際には、このようなフィルタを通します。海に近いので、塩害防止のために除塩フィルタをこの奥に設置しているそうですが、こちらは予想外に塩が入り込まないとのこと。

窓からの眺めは最高です。取材に訪れた日は晴れていたので、手稲山までハッキリ見えます。現在2号棟まで完成している石狩DCですが、8号棟まで建てられる分の用地を取得済みとのこと。実に広々としています。



そして、建物周辺の私道はDCからの排熱でロードヒーティングをしています。



共連れ防止のゲートをくぐった先は、いよいよサーバー室です。



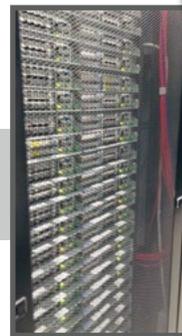
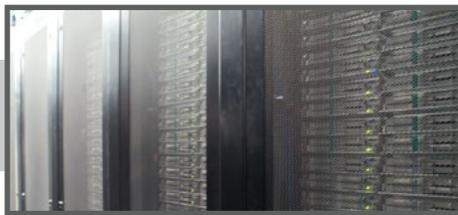
まず目に飛び込んでくるのがHVDC DC12Vシステムのサーバーラックです。



※1 PUE(Power Usage Effectiveness): データセンターやサーバー室のエネルギー効率を示す指標の1つ。データセンター全体の消費電力をサーバーなどのIT機器の消費電力で割った値で、これが1.0に近づけば近づくほどエネルギー効率がよい。通常の都市型DCではPUEが2前後。



1Uハーフサイズのサーバーが多数稼働しています。これらはさくらの専用サーバーで用いられているものだそうです。



皆さん! LinuxやUNIX好きな皆様がハアハアする物を見つけてしまいましたよ!

こちらはサーバー室の電源ラックなのですが、ラックの前に立つだけで分電盤などの状況をモニターで見ることができるのです。え?それだけ?と思われるかも知れませんが、なんとこれ、GNOMEのアプリケーションなんです。(※2)



そして、この自動扉の向こうは2014年2月より運用開始した2号棟サーバー室です。

1号棟の扉は手動だったのですが、田中社長自らサーバーを運んでラッキングしに来たときに「扉は手動よりも自動がいいよね。」と言ったのが自動扉を採用したキッカかだそうです。写真には写っていませんが、廊下からサーバー室は段差がなく、台車でサーバーを運んでも荷崩れする心配がありません。



宮下センター長は「1号棟では予想以上に排熱効率や電力効率が良かったので、2号棟はもっとラックをギチギチに詰めなよと田中社長からプレッシャーをかけられました。」と語っておられました。



最後に見せていただいたのは、休憩室です。テーブルエリアと畳エリアがあり、お菓子やパン、飲み物の販売もあります。大雪で帰れなくなったときはここで夜を明かすこともあるそうで、まさに北の大自然、試される大地といったところでしょうか。



※2 GNOME: X Window上で動くデスクトップ環境。http://www.gnome.org/

一通り施設を見学したところで、 会議室にてインタビューを 行いました。



濱田:2014/11/15に石狩DC開所3周年を迎えますが、感想を聞かせてください。

宮下:(開所から3周年を迎えたのが)早いなーという印象と、大きなトラブル無くやってこれたのが良かったと思います。他のDCではやっていないであろう新しい試み、例えば外気冷房を取り入れたりですとか、そうした試みを導入しつつも大きなトラブルが無くて良かったです。あとは1号棟が早く埋まってよかったですね(笑)。本当は3年で埋まることを目標としていましたが、2013年に2号棟の運用を開始することができました。

濱田:通常、DC運用は目新しいことをあまりやらず、決まったフローに忠実なイメージがありますが、新しいことをやりつつも無事故という品質を保っているのには、どういった工夫をされているのでしょうか？

宮下:サーバー運用ですと、ヒヤリハットを挙げて日々改善したりですとか、設備設計ですと、業者さんと一緒に実験したり導入検討したりして進めています。設備担当の者が元々建設業界にいたので、そういう知識をもとに弊社が主体となって、業者さんと協力して進めていったので大きな事故に繋がらなかったのかなと思います。全てにおいて当事者意識を持って一人称で取り組んで、全てにおいて業者さんに投げっぱなしにしないというやり方でやってきています。

濱田:日々の運用オペレーションでどのように運用改善などに取り組まれていますか？

宮下:先ほど申し上げましたヒヤリハットの他にも、いろんな各種サービスを扱っていますので、サービスごとに手順があるんですけども、手順は作りっぱなしにせず常に改善するようにしています。どれだけ改善しているのかは定量

的に測っています。手順改善してくれた人を表彰するような制度も設けて、自発的に改善していく仕組みを作っていますね。

濱田:それはボトムアップ的なアプローチでしょうか？

宮下:そうですね。ボトムアップで運用改善していくよう、ここ数年取り組んでいます。今年からは私のいる運用部に関しては、各エリアにマネージャーを置いて、各拠点にはリーダーを置いて、各エリアで決済が早くできるようにしましたし、東京に複数ある各DCにリーダーを置きましたので、センターごとに素早く反応できるというような仕組みを作りました。

濱田:サーバー構築の自動化などの仕組みはありますか？

宮下:ツールの開発はだんだん出来始めていますが、正直なところ新しいことをやりながら運用部での開発は厳しいですね。ansibleなどもやりたいですし、個人的に始めているのですが、現在BIOSやIPMIの設定は手動でやっているのですが、今後は物理的なハードウェア設定を自動化したいと思っています。メーカー製のツールを中心に探しているんですけど、メーカーが提供しているツールでは実現できない機能は、キー入力の再生ツールなどで多台展開してみたいなどは思っています。

濱田:最近ですとスイッチやルーターも物理的な結線の制約から開放されつつありますが、ネットワーク設計や運用の自動化も検討されていますか？

宮下:レイヤ2スイッチについては、APIが搭載されているものはこれにアクセスして設定することで自動化しています。あとはDOS攻撃の検知も自動化しています。今までは手動オペレーションで30分くらいかかって対応していたのが、自

開所3周年の意気込み



動化によって数分でできるようになりました。こうしたオペレーションの自動化というのは各部署でキーワードになっていますね。

濱田：今年のRISING SUN ROCK FESTIVAL 2014 in EZO(※3) (以下RISING SUN)で公衆無線LANを提供されましたが、参加のキッカケは何だったのでしょうか？

宮下：去年のさくらの夕べ@札幌の懇親会で、そんな話がありました…

三谷：そうですね、澤田(local代表理事)が直訴しにきました(笑)

宮下：私もRISING SUNは個人的に見に行っていて好きでしたし、前の年に会場に行ってTwitterでつぶやくにも時間がかかるという経験をしましたので、そのアイデアをもらってやってみようと思ったのがキッカケですね。

濱田：アンテナ以外は自前で建設されたそうですね。

宮下：アクセスポイントに関しては札幌の会社さんに協力してもらいましたが、設定やアンテナまでの配線は自分たちでできました。来年やれるとしたら、もっと自分たちでできる率を増やしたいですね。

濱田：会場まで結構な線路長がありますよね？

宮下：石狩DCから会場までは300mあります。センターと会場間は無線ですが、センターのサーバー室から屋上のアンテナまでは90mほどLANケーブル配線をしました。今までは屋内のケーブル配線しきしないので、屋外用のLANケーブルは硬くて重たいなと思いながら配線した思い出がありますね(笑)

三谷：いつも仲間がRISING SUNに行くのを家からTwitterで見ているのですが、去年までは『行ってきます』から『ただいま』まで様子が伝わってこなかったのですが、今年はすごかったですね。みんながツイートしたりFacebookに写真を載せたりするので、リアルタイムで様子がわかりますね。事情に詳しい人は『さくらさんのおかげだ』と言ってました(笑)。RISING SUNに行った人も行けなかった人も、リアルタイムで楽しむことができましたね。

宮下：今年イベントーさんとの接点もできたので、今後拡大できたらなーと思います。

濱田：2015年夏のRISING SUNに参加する予定はありますか？

宮下：桜井とは社内的な作戦どうするか、イベントーさんでどういうアプローチをしていくかの作戦会議を先日行いました。

室野：やらないという選択肢はまったくくないですね。

宮下：こうしたイベントへの協力を、社内実地研修や新人研修の場にもできたらいいですね。4月入社の子が構築に携わって夏8月に卒業できるみたいなのも面白いと思っています。あとは、さくらに入社するとこういったイベントの仕事にも参加できるというメリットも押し出して求人にもつなげていきたいですね。

濱田：RISING SUNのときに一番苦労されたのはどんなことでしょうか？



※3
RISING SUN ROCK FESTIVAL 2014 in EZO: 毎年夏に石狩湾新港 野外特設ステージで開催されている国内最大級の野外オールナイトロック・フェスティバルイベントの1つ
<http://rsr.wess.co.jp/2014/index.html>

宮下:特に目新しいことをやったわけではないのですが、業者さんの手配とか、どういう設計にしようかとか、(屋外ネットワーク構築は)初めてのことだったので、手探りでした。はじめはセンターから会場まで光ケーブルを敷設しようという案が出たのですが、見積もりを取ったら10日間で数百万円かかると言われてしまいました。なんとかならないかと思っていたところに無線にたどり着きました。屋上のアンテナマストも最初は自分たちで建てようと思っていたのですが、高さもありますし、風があることも考えると、作業時の安全という観点で、最初の年は自分たち素人じゃ無理だなーと。こうした工事をしてくれる業者さんを見つけるのに苦労しましたね。中のネットワークやルーターの設定はいつもの延長線上だったので問題なくできましたし、監視などもいつものようにできました。

濱田:石狩DCへの見学は今もよく訪れる人はいるのでしょうか？

宮下:開所当初よりは減りましたが、他のDCよりは多いですね。月に2〜3件くらいですね。利用を検討されている方が半分、取材が半分ですね。運用中の苦労話はよく聞かれます。何人くらい働いていますかというの聞かれますね。

室野:現地採用はありますか？という質問もよくされます。

宮下:自治体さんなどのような、データセンター誘致を検討されている方は、雇用の話は気になるみたいですね。利用を検討されている方は運用に関する質問が中心ですね。あとは周囲に何もないので、従業員のお昼ご飯はどうしてるんですかということも聞かれます(笑)。

濱田:こちらへんセイコーマートくらいしかないですね・・・やはりお弁当買って持ってくる人とかが多いのでしょうか？

宮下:11時くらいになると『お弁当屋さんがありました』ってアナウンスが流れるので買いに行く人もいますね。周囲に工場があるので、お弁当さんの巡回コースになっているんですよ。周囲に工場もなくてうちだけだったらお弁当さんが来てくれるかどうか・・・(苦笑)

濱田:オペレーターの訓練や教育は、どのようにされていますか？

宮下:石狩DCで働くオペレータの教育に関して言えば、物理的な作業が中心ですので、ハードウェアの仕組みやRAIDについて、IPMIについてなど、ハードウェアに関する教育メニューを中心に行っています。あとは物理的な結線やラック内のエアフローの基礎知識、電源コンセントの種類についても触れますね。特殊な形状のコンセントも扱ってますので。意外に思われるのですが、在庫の処理については教育に時間をかけています。帳簿をつけるのに在庫管理のシステムを使っているのですが、これのオペレーションが意外と難しいので。あとはサーバー構築に関しては、自

DCの設計は作業員の効率や健康を考えます



動化は進んでいるんですけど、一部手動でファイルを設定しなければいけない箇所があるので、viの使い方を教えたりですとか、UNIXの基本コマンドを教えたりはしていますね。

濱田:サーバー構築でのオペレーションは、自動化が進んでいるということから、確認のコマンドを実行することが多いのでしょうか？

宮下:確認の工程もありますし、さくらの専用サーバーは基本セットを一旦構築するのですが、お客様からオプションのオーダーがあったりすると、一旦サーバーをラックから外して、例えばストレージをSSDに変更したりなどのカスタ



マイズを行います。こうした際にファイルを編集することはあります。あとは障害対応でファイルを編集したりということもありますね。

濱田：ネットワークのオペレーションは自動化などを行っていますか？

宮下：うちのチームですが、ネットワークのオペレーションは物理的な配線が中心で、基本的なconfig投入までを行います。それ以降は運用チーム(東京や大阪)に任せています。



ITを使って音楽フェスを盛り上げた。 きっかけは懇親会



濱田：最後に、『こんなサービスを予定している』『こんなサービスをやりたい』『こんな石狩DCにしたい』といったことはありますか？

室野：IaaSの範疇を超えるサービスにまで手を広げるという予定はなく、あくまでインフラに関するサービスを安価に提供することを大前提にしています。

宮下：AWSさんは大小含めて200以上ものサービスがありますが、さくらはそういった追求はせず、範囲を絞って開発してこうという話は代表(田中社長)からもあります。石狩DCに関しては、自社の設備だからこそできることがいっぱいあるのですが、例えば空調ですとか、HVDCなどがありますので、こうした最新の設備を常に取り入れられるようなデータセンターにしたいです。日々のオペレーションに関し

ても、物理的な作業がメインですが、それに伴う運用ツールの開発を自分たちでできるようにしたいです。最近のエンジニアの方って、WEBのフロントエンドですとかソフトウェア会社のインフラエンジニアでもクラウドに移行して物理サーバーを見る機会が減っていますけれども、幅広いレイヤーで深く対応できるエンジニアを育てていきたいですね。

三谷：空調の壁吹き出しと天井吹き出しの2種類を同時に採用した理由を教えてくださいませんか？

宮下：実験が契機ですね。壁吹き出しはあまり例がないのですが、導入当初は、空気の均一性が悪いのではないかと懸念していました。一方で、天井吹き出しは空気の流れを作るためにダクトを作らなければならないという手間と部材の費用がかかります。天井吹き出しの空調機(ファン)に関しては、遠くまで空気を運ぶので強力なファンにしなければいけないので、電気代もインシヤルコストもかかってしまいます。反面、壁吹き出しにするとダクトも不要で、ファンのモーターも小さくでき電力もかかりません。それぞれメリット・デメリットがあるので、両方導入してみようということになりました。導入していろいろと計測してみると、空調能力は壁吹き出しのほうが均一性がなくなるという懸念はありましたが、各サーバーラックにセンサーを入れて温度を測ったら、そんなにバラついておらず、両方でそんなに空調能力に大差はなく、差異は許容できる範囲でした。ただ、壁吹き出し方式は人に直接風が当たってしまいますので、長時間作業が辛いです。ラッキング作業は1回に何十台と行いますので、短くて半日、通常1日作業になってしまい、作業員の効率や健康を考えると天井吹き出しのほうが良いだろうという傾向です。なので、ほとんどのラックは天井吹き出しにしています。ただ、壁吹き出しを諦めるのは勿体無いねって話もありまして、風が直接人に当たらないようにフィルタをラックとラックの間に入れてみたりというような実験をしています。コスト面でも壁吹き出しのメリットもありますので。

濱田：本日はどうもありがとうございました。



取材
協力

三谷 公美 / Kumi Takahashi MITANI
(一般社団法人 LOCAL)

宮下 頼央 (さくらインターネット)

室野 奈緒美 (さくらインターネット)

シェルスクリプト大喜利

第十五回

司会：『もっと吹く』編集長・みかん

皆さんこんにちは。三か月に一度のお楽しみ、シェルスクリプト大喜利のコーナーのお時間です。三か月に一回のはずが、出題の不幸で連載が一月遅くなってしまいました。ごめんなさい。

もたもたしてるうちに年が明けてしましまして、今年^{ひつじ}は未年なんですってねえ。ヒツジといえば鳴き声は「メェ〜」ですが、そういえばヤギも同じだったような……。ってことで、Webで検索したら鳴き声と比較する動画があったんです。ヒツジは「メー」でヤギは「メェ」と鳴いてました。いやあ、12年前はこんな簡単に疑問解決できたかな？

さてそれでは本コーナー、シェルスクリプト大喜利のルールのご説明からまいります。

シェルスクリプト大喜利とは

シェルスクリプト大喜利特有のルール

- sh大喜利はクイズやテストではありません。なので決まった答えというものはないのです。あえて言うなら面白いスクリプトが正解！
- 面白いスクリプトとは、例えばこんなもの。
 - イ、人が考えつかない意外性がある
 - ロ、美しい、芸術的、記述がシンプル、高速、など
 - ハ、アイデア・こだわりが光る
 - ニ、ネタになるバカバカしさ、くだらなさがあるなど。でも最後のは段位強制返還の恐れありよ。:-)
- スクリプト動作環境はLinuxとし、基本的にLinux JM(<http://linuxjm.sourceforge.jp/>)に記載されているコマンド及び機能のみ使用可能とします。これは多くの人を楽しめるようにするためなのです。(JMにあるので、Cシェル系での解答もOK！ あどksh、zshもOKにひまむた)
- sh大喜利はシェルスクリプトを披露する場なの

で、PerlやRuby、Pythonなどは使っちゃダメです。そもそもJMにも載っていません。逆にシェルスクリプトにとって不可欠なawkやsed等はOKです。JMにもありますし。でも、よっぽど面白ければ、Perlとかなきにしもあらず？

五、Open usp Tukubai(<http://uec.usp-lab.com/>)も使用OK！ 但し、使う意義がそれなりに感じられないと採用はキビシいですよ〜。

ルールもおさらいしたところで、さあ始めましょう！

本番開始

<一問目>

1 ~ n (nは引数で指定) の自然数の中に存在する『回文数』を全て求めるシェルスクリプトを書いてください。

年が明けてもやっぱり一問目は整数問題です。

今回は回文数です。つまり、各桁の数字が左右対称になっている数のことです。以前、回文を判定するお題を出していただけに、ちと簡単だったかな？

◎K 師範の解答

```
1 #!/bin/sh
2
3 seq $1 |
4 while read n; do
5   echo $n | fold -w 1 > $$$.1
6   tac $$$.1 | diff $$$.1 - > $$$.2
7   if [ ! -s $$$.2 ]; then
8     echo $n
9   fi
10 done
11 rm $$$.?
```

まずはヒネリがないと謙遜するK師範の解答から。

いやいや、回文数を作るのにfoldで縦並びにしてtacで反転するあたりはアイデアだね。しかもdiffの出力が無し(=等しい)なら回文数と。師範、いつもありがとう！ 一段落と、只今四段。

◎ぞよかせさんの解答

```
1 #!/bin/sh
```

```
2 awk -v n=$1 'BEGIN{for(i=1;i<=n;i++){print i}}' |
awk '{for(i=length($0);i>0;i--){printf("%s",subst
r($0,i,1))}print " ",$0}' | awk '1==2{print $1}
scale=0
```

初投稿のそよかせさん。こちらは実質ワンライナーですな。

全部 AWK か！最初の AWK で数列作って、次ので逆並びの数を作り、最後ので回文判定をしてるわけね。最初の AWK なんかは seq コマンドでもっと短く書けるけど、それはきっと AWK だけで済ませようというこだわりだろうね。ありがとう。初級検与。

◎ふじやんさんの解答

```
1 #!/bin/bash
2 seq $1
3 sed 's/.*/ &\n&/'
4 sed '/^[0-9]/s/.*/echo & | rev/e'
5 sed 's/ //g'
6 uniq -c
7 sed -n '/2 /{s/.*/ //;p}'
```

一転して今度は、sed がいっぱい並ぶ解答です。まあ、2 回目の投稿ですね、ありがとうございます。添付のコメントによれば「uniq コマンド以外は全部 sed で頑張りました。嘘です。頑張っただけです」とのことです。なんじゃそりゃ！

ぱっと見は uniq 以外 sed に見えるけど、いやいや最初は sed じゃなくて seq だし、途中で echo や rev が出てくるしいろいろ使ってるじゃないか。

でもアイデアはいいね。1 個目の sed で値をもう一行複製して、次の sed でその片方を rev コマンドによって反転させ、最後の uniq と sed で連続出現したもの（=回文数になっているもの）を見つけて残しているわけだ。面白い、一級検与。只今二段。



では、二問目いきましょう。

<二問目>

横 70 桁×縦 70 行でハイフン “-” が敷き詰められ、そのうちのいくつかがアスタリスク “*” で書き換えられたテキストがあります。これを入力データとし、マインスイーパーのように、自分のいるマスの周囲 8 マスにある “*” の数を数えて 7 個以上であれば数字に置き換えるシェルスクリプトを書いてください。

```

- * - - - - -
- * - - - - -
- - - - - * - - -
- - - - - * - - -
- - - - - * * * -
- - - - - * * * -
- - - - - * * * -
- - - - - * * * -
```

↓

```

- 1 * 1 - - - - -
1 2 2 1 - - - - -
1 * 1 - - - - -
1 1 1 - 1 2 2 1 -
- - - - - 1 * * 1 -
- 1 1 1 - 1 2 2 1 -
- 1 * 1 - 1 2 3 2 1
- 1 1 1 - 2 * * * 2
- - - - - 3 * 8 * 3
- - - - - 2 * * * 2
```

今のようにバラエティーに富んだスマホゲームなど無かった頃、あるいは Windows や Linux に GUI 環境を入れたばかりの頃、よくやりませんでしたか、マインスイーパー（もどき）を。プレイして、爆弾に触れてゲームオーバーになると、各マスに周囲 8 方向に隣接する爆弾数がパッと表示されますよね。アレがやりたいんですよ。

行を跨いだカウントをどうやるかが腕の見せ所のお題。さて、どんな解答が来たのでしょうか。

◎fkd.sh さんの解答

```
1 #!/usr/local/bin/bash
2 # usage: $ bash maine.bash < maine.txt
3
4 declare -A A C
5
6 for i in {1..10}; do
7 read A[$i:1] A[$i:2] A[$i:3] A[$i:4] A[$i:5] A[$
i:6] A[$i:7] A[$i:8] A[$i:9] A[$i:10]
8 done
9
10 for i in {1..10}; do
11 for j in {1..10}; do
12 if [ "${A[$i:$j]}" = "*" ]; then
13 C[$i:$j]="*"
14 else
15 if [ "${A[$((i-1)):$((j-1))]}" = "*" ]; then
16 let C[$i:$j]++
17 fi
18 if [ "${A[$((i-1)):$j]}" = "*" ]; then
19 let C[$i:$j]++
20 fi
21 if [ "${A[$((i-1)):$((j+1))]}" = "*" ]; then
22 let C[$i:$j]++
23 fi
24 if [ "${A[$i:$((j-1))]}" = "*" ]; then
25 let C[$i:$j]++
26 fi
27 if [ "${A[$i:$((j+1))]}" = "*" ]; then
28 let C[$i:$j]++
29 fi
30 if [ "${A[$((i+1)):$((j-1))]}" = "*" ]; then
31 let C[$i:$j]++
32 fi
33 if [ "${A[$((i+1)):$j]}" = "*" ]; then
34 let C[$i:$j]++
35 fi
36 if [ "${A[$((i+1)):$((j+1))]}" = "*" ]; then
37 let C[$i:$j]++
38 fi
39 fi
40 done
41 done
42
43 for i in {1..10}; do
44 for j in {1..10}; do
45 echo -n "${C[$i:$j]}:="
46 done
47 echo ""
48 done
```

まずはお一人目、初投稿ありがとうございます。な

るほど、配列変数に一旦全部読み込んでから、全マスについて周囲の爆弾数を数えるというわけですね。オーソドックスにきましたね。

おお、よく見たらこのシェルスクリプトは全て bash の内部コマンドで閉じてるぞ。つまりピュア bash ってことか。うーん、やるね～！よし、**初段授与**。

◎TSB_KZKさんの解答

```
1 cat mine.txt |
2 tr '*-' '50' |
3 sed "s/./& /g" |
4 awk '
5     NR>1{
6         for(i=1;i<=10;i++){
7             if(p[i]>=5) n[i]=1;
8             if($i>=5) p[i]+=1;
9             printf("%d ", p[i])
10        }
11        print ""
12    }
13    {
14        for(i=1;i<=10;i++){
15            p[i]=$i+n[i];
16            n[i]=0
17        }
18    }
19    END{
20        for(i=1;i<=10;i++){
21            printf("%d ", p[i]);
22        }
23        print ""
24    }' |
25 sed "s/ //g" |
26 sed "s/^\(echo /g; s/\([0-2]\)\[56\]/\((\1+1)\)5/g" | sh |
27 sed "s/^\(echo /g; s/[56]\([0-3]\)\[5\]/\((\1+1)\)/g" | sh |
28 tr '056789' '*****'
```

続いては、常連で本誌連載でもお世話になっている TSB_KZK さん。お世話になってるんですが……、いやあ～これはオシイ!! 不正解。

とりあえず添付のコメントを紹介すると「awk で縦の計算、sed で横の計算をしました」とのこと。つまり斜めの計算が足りないのですよ。でも、awk で縦方向、sed で横方向という分担が面白かったのと、sed でシェルに食わせるコードを動的に生成しているトリッキーさが勿体無かったので掲載しましたよ。

うーん、残念！変わらず七段。

◎C₆H₈O₇ (クエン) さんの解答

```
1 cnt_bottomright='
2 {
3     for (i=1; i<=NF; i++) {
4         $i=(p[i-1]=="*")&&($i!="*")?$i+1:$i;
5         $i=(p[i ]=="*")&&($i!="*")?$i+1:$i;
6     }
7     for (i=1; i<=NF; i++) {p[i]=$i;}
```

```
8     print;
9 }'
```

```
10 cat mine.txt
11 tr :- :0
12 awk "$cnt_bottomright"
13 ./tateyoko
14 tac
15 awk "$cnt_bottomright"
16 ./tateyoko
17 tac
18 awk "$cnt_bottomright"
19 ./tateyoko
20 tac
21 awk "$cnt_bottomright"
22 ./tateyoko
23 tac
24 tr 0 -
```

こちら常連のクエンさん。Tukubai の tateyoko コマンドを使ってますね。コメントによると、爆弾の真下と右下をカウントするという作業を 90 度ずつ回転させながら 4 回行って、周囲 8 マスのカウントを実現しているとのことですよ。

なるほど、そういう手もあるのか！しかもコレ、縦横のマス数が 10 でなくても使えるや。すばらしい！**二段授与**だ。これで七段。



では最後のお題。今回はフツーなお題なんです。

<三問目>

UNIX の cal コマンドは日曜始まりでひかありません。そこで、引数で 0 (日曜) ~ 6 (土曜) の数値を与えると、その日曜始まりで今日のカレンダーを表示するシェルスクリプトを書いてください。

世界は広くて多種多様！カレンダーにも金曜始まり、土曜始まりなど、日曜が始まりでない地域がたくさんあります。シェルスクリプトの力で i18n (国際化) しようじゃありませんか！とうのがこのお題。

実は今回一番人気がありました。解法も様々……。やっぱりフツーのお題の方がいいのかなあ。

◎さりたま帝国さんの解答

```
$ n=ここに曜日番号
$ cal | tail -n +3 | tr -d '\010\137' | sed 's/\(..\)/\1_/g' | sed 's/[ _]*$//' | tr ' ' '0' | tr _ \
\n | awk '/00/{w++;next} /01/{for(i=0;i<(7+w-$n)%7;i++){print "--"}1}' | awk 'NR%7{printf("%s ",$1);next} 1' | grep ^
```

最初はオーソドックスな解法の解答から。ワンライナーですね。ええとこれは cal コマンドの日付を一旦 1 つ 1 行にして、再び 7 日毎に 1 行とするもの。ただし、何日目を起点にするかは指定された曜日番号に従うというわけですよ。

まあそつない解答だね。ただ、最後の grep が何のためにあるのかちょっと気になったけど、改行コードを入れるためなのか。なるほど。**一段撥与**。只今五段。

◎321576さんの解答

```
1 n=ここに曜日番号
2 mdate -d $(date '+%Y%mm')
3 tarr
4 yobi 1
5 self 1.7 2
6 awk 'NR<2{for(i=0;i<($2-'$n'+7)%7;i++){print "***}}}'
7 self 1
8 yarr -7
9 tr '* ' ' '
```

次も一日毎に一旦1行にする戦略をとってるんですけど、他者と違って cal コマンドを使ってないのが注目ポイント。Tukubai コマンドを駆使しているんですが、mdate という日付を計算・列挙するコマンドで今月の日にち一覧を出してます。

なるほどね、mdate コマンドってこういう列挙する用途に向いてるのか。年月によって月末の日にちは28～31日と様々で、計算面倒だからねえ。よし、これも**一段撥与**。只今五段。

◎fkd.shさんの解答

```
1 #!/usr/local/bin/bash
2
3 # usage:
4 #$ bash cal_youbi.bash 0
5 #$ bash cal_youbi.bash 0 12 2014
6
7 OF1=$(( ${1:-0} * 2 + 1 );
8 OF2=$(( ${1:-0} * 3 + 1 );
9 paste -d " " \
10 <(cal $2 $3 | awk 'NR == 2{print $0; print "
    " NR != 2{print $0}') \
11 <(cal $2 $3 | awk 'NR == 1{print ""}NR>1') | \
12 awk -F, 'NR == 1{print $0}NR == 2{print substr($
    1, ${OF1}, 13)}NR > 2{print substr($1, ${OF2}
    ', 21)}' | \
13 awk 'NF>0'
```

誌面が無くなってきたので最後にしますけど、これは面白い解法ですよ。

「なるほど、頭一」とアタクシ感心したので、解説をしちゃいます。9、10、11行目で同じ月のカレンダーを横に並べてるんです。ただし左側(10行目で生成している方)は日にちを1行下にずらしておきます。するとこういうテキストができます。

February 2015							同じ月を、1行ずらして横に並べる						
Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7	8	9	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
22	23	24	25	26	27	28							

金曜始まりにしたいなら金曜の列から7列抽出

そうしたらあとは、指定された曜日の始まりに応じ

て必要な7列分を取り出せばいいというわけです。

いや～、これはウマイ!! アタクシ今ねえ、**ねるねるねるねの魔女**みたいに脳内がテレってるよ! (←古いつ)

ただねえー、残念なことに曜日名の行が正しく取り出せてないのだよ!!! おしい! 曜日名の行の取り出し開始位置を7行目で指定しているようだけど、これ、8行目(日にちの列)と同じでよかったんじゃないか!? さらに欲を言うと、LANG=Cを設定してCロケールにしておかないとこれまた文字数を数え間違う原因になってしまうぞ。

でもまあ、お題の解釈のしよによっては曜日名の行は出さなくてもいいともとれるし、何よりアイデアが気に入ったから、**二段撥与**しちゃう。これで三段だ。



といったところで本日の大喜利はこれにてお開き! 読者の皆さん、投稿してくれた皆さん、今回もありがとうございました。

ありがどうございまして!

これにておしま!

約四年にわたって開催してまいりましたこのシェルスクリプト大喜利。思えば大喜利ネタでマウスパッドを作ったり、大喜利の出題ミスを解答でツッコまれたり、想像の上(時には斜め上)行く解答で勉強になったり、いろいろありました。

しかし突然なのですが、今回をもちまして御開きとさせていただきます。ここまでお付き合いいただいた皆様、大変ありがとうございました。また、投稿をお寄せくださった皆様、せっかくここまで段位を重ねてくださったのに、ごめんなさい。

というのもこのところ、シェルスクリプト界隈でナゾの秘密結社が暗躍してるって噂で……。コイツは大変だ! ってことでアタクシも黙っちゃいられなくなっちゃいましたね。火事と喧嘩は何とやらって奴です。

でもまたやりたいねえ

でもこの一件が落ち着いたら、また読者の皆さんを巻き込んで同じような企画をやりたいですね。

その時まで、シェバンしのお別れ! …………… ええ～と、はい、おあとがよろしいようで。(寒う)



HANDS LAB

ユニケージ®エンジニア数 **最大級!**

ハンズラボはユニケージ®開発手法に特化したITソリューション企業です。東急ハンズの営業システムを刷新したノウハウを駆使し、小売業における「オーダーメイド」のシステム開発を行います。

中途採用 大募集! 詳しくはHPで!

> <http://www.hands-lab.com/>

「ユニケージ®」は有限会社ユニバーサル・シェル・プログラミング研究所の登録商標です。

これであなたもユニケージエンジニア!

ユニケージ開発手法教育講座

「ユニケージ開発手法教育講座」は、ユニケージ開発手法におけるデータ管理の方法や、オリジナルコマンドの使用方法などをハンズオン形式で具体的に学べる講座です。UNIXの基礎からユニケージ開発手法による開発プロジェクトの進め方まで、ユニケージエンジニアとしてのトータルスキルを習得できます。

<http://www.usp-lab.com/LECTURE/CGI/LECTURE.CGI>



- K-BASIC** ユニケージ基礎編
- K-WEB** WEBアプリケーション編
- K-BATCH** バッチ処理編 (ウェブアプリケーション処理)
- K-ARCH** ユニケージアーキテクチャ編
- K-SETUP** ユニケージ開発環境セットアップ編
- K-UNYO** システム運用・管理編
- K-PROJECT** プロジェクトマネジメント・人材育成編
- K-SQL** 速習: SQLからの移行編
- K-STAT** ユニケージにおける統計コマンド編

続々と新講座も充実中!

UNIX 初心者のための講座などもご用意しています。

TechLION
For Independent Engineers

技術の草原で百獣の王を目指す
エンジニアたちの新感覚トークライブ!

<http://techlion.jp/>

上記のサービス内容は2015年1月現在のものです。最新の情報はホームページにてご確認ください。

ISBN 978-4-904807-16-3
C3455 ¥500



9784904807163

USP 研究所
定価 (本体 500円+税)



1923455005002