**Zajęcia 10 - „Polecenia shella - skrypty - zmienne specjalne w skryptach, polecenie read, funkcje, debugger”.**

**Sprawozdanie 5.**

Wykonał(-a): ……………………….

Grupa (ISI/IO, nr) : ………………..

**10.** Przygotować: Podsystem plików – powłoka, skrypty: zmienne specjalne, read, funkcje, debugger.

Na zajęciach: skrypty -wykorzystanie zmiennych specjalnych, polecenia read w skryptach. Tworzenie funkcji w skryptach. Wykorzystanie debuggera do weryfikacji wykonania skryptów, polecenia: eval, echo, exit, listy warunków, rozwinięcia parametryczne.

Ćwiczenia: SO Linux – skrypty: zmienne specjalne, read, funkcje, debugger; polecenia: eval, echo, exit, listy warunków, rozwinięcia parametryczne.

Sprawozdanie 5.

**Sprawozdanie.**

**ZADANIA ĆWICZENIOWE**

**1)** Wykorzystanie zmiennych specjalnych.

Skrypt pobiera w trybie wsadowym pewną ilość parametrów – w zależności od wariantu -

wybór wariantu: (((21 +7)\*3) mod **6** )+1:5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **War.1** | **War.2** | **War.3** | **War.4** | **War.5** | **War.6** |
| 4 | 5 | 4 | 3 | 5 | 4 |

, następnie:

- wyświetla nazwę skryptu,

- podlicza i wyświetla inf. o tym – ile parametrów zostało pobranych oraz je wyświetla,

- jeżeli ich liczba jest mniejsza niż wymagana – stosowny komunikat i zakończenie skryptu z kodem 31,

- dla war. 1, 3 – sprawdza – czy istnieje w bieżącym katalogu plik o nazwie jak 3-ci parametr – jeżeli jest – skrypt go usuwa, tworzy podkatalog o nazwie jak 1-szy parametr;

- dla war. 2, 4 – sprawdza – czy istnieje w bieżącym katalogu plik o nazwie jak 3-ci parametr – jeżeli jest – skrypt wyświetla jego zawartość, tworzy nowy plik o takiej samej nazwie – zapisując do niego zawartość katalogu bieżącego;

- dla war. 5-6– sprawdza – czy istnieje w bieżącym katalogu plik o nazwie jak 2-gi parametr – jeżeli jest – skrypt zmienia mu nazwę na nową – jaką zawiera 3-ci parametr.

#1/bin/bash

echo $0

echo "ilosc parametrow $#"

echo $\*

if (( $# < 5)); then

echo "podano mniej niz 5 argumentow"

exit 31

fi

if [ -e "$2" ]; then

mv "$2" "$3"

echo "wykonano zamiane"

else

echo "plik '$2' nie istnieje"

fi

**2)** Wykorzystanie polecenia read.

Skrypt po uruchomieniu wyświetla inf. o tym, że rozpoczął działanie, następnie wyświetla nazwę katalogu bieżącego, potem - inf. o tym – jaki warianty działań są do wyboru, następnie – wczytuje opcję wybraną przez użytkownika i wykonuje stosowne dla tej opcji polecenia Opcje zależą od wariantu – Tabela 1. Nazwy stosownych zasobów podawane są po wybraniu stosownej opcji – czyli polecenie read.

Wybór wariantu: (((21 +4)\*5) mod **3** )+1. 3

Tabela 1

|  |  |  |  |
| --- | --- | --- | --- |
| Działanie: | **war.1** | **war.2** | **war.3** |
| - wyświetlenie inf. o zawartości podkatalogu o podanej nazwie | \* |  |  |
| - wyświetlenie zawartości pliku o podanej nazwie |  | \* |  |
| - utworzenie pliku o podanej nazwie – z zapisem do niego inf. o wszystkich zarejestrowanych w systemie użytkownikach – ich loginy i UID |  |  | \* |
| - sprawdzenie – czy plik „Kopia1” jest w katalogu o podanej nazwie | \* |  |  |
| - sprawdzenie – czy podkatalog o podanej nazwie jest w bieżącym katalogu |  |  | \* |
| - skopiowanie pliku o podanej nazwie do innego – też z podawaną nazwą iw tym samym katalogu |  | \* |  |
| - sprawdzenie – czy grupa użytkowników o podanej nazwie jest zdefiniowana w systemie, jeżeli tak – to jaki jest jej GID | \* |  |  |
| - sprawdzenie – czy użytkownik o podanej nazwie jest zalogowany w tej chwili do systemu, jeżeli tak – to jaki jest jego UID |  | \* |  |
| - sprawdzenie – czy są pliki, które zawierają wiersze zaczynające się na podaną frazę |  |  | \* |

**3)** Napisać skrypt, który wykonuje stosowne działania – wybór wariantu: (((<dzień urodzin> +4)\***4**) mod **6** )+1.

War.1 Przy uruchomieniu skryptu jako pierwszy parametr podajemy nazwę pliku – jeżeli jest ona równa nazwie wykonywanego skryptu to skrypt wykonuje następujące czynności:

- wyświetla informację o wszystkich plikach z katalogu domowego, mających uprawnienia rwx dla właściciela, r dla grupy i zapisuje je do pliku o nazwie podanej jako drugi parametr skryptu,

jeżeli nie to:

- wyszukuje wszystkie pliki specjalne – tzw. urządzenia znakowe, wyświetla informację o nich i podlicza – ile ich jest.

War.2 Przy uruchomieniu skryptu podajemy przynajmniej 3 parametry. Pierwszy parametr ustala, która z opcji skryptu ma być wykonana - jeżeli pierwszy parametr jest równy:

- „A1” - to skrypt wyszukuje nazwy użytkowników zdefiniowane w systemie oraz ich UID

- „A2” – to skrypt tworzy plik w bieżącym katalogu o nazwie podanej jako drugi parametr z prawami dostępu rw-r--r--, do którego zapisuje nazwy wszystkich plików z bieżącego katalogu;

- „A5” – to skrypt tworzy plik w bieżącym katalogu o nazwie podanej jako drugi parametr i zapisuje do niego informację - nazwę bieżącego katalogu.

War.3 Po uruchomieniu skrypt wyświetla on inf. o zawartości bieżącego katalogu, następnie podajemy mu nazwę katalogu. Skrypt wyszukuje pliki z podanego katalogu i jego podkatalogów do poziomu 2, zawartość których była modyfikowana nie później (mniej) niż przed 2 dniami. Ich nazwy są zapisywane do pliku o nazwie „Zmiany” w katalogu domowym. Skrypt przeszukuje ten plik i ustala – ile wśród nazw jest zaczynających się na jedną z liter: od b do p.

War.4 Po uruchomieniu skrypt wczytuje nazwę katalogu – jeżeli jest to katalog bieżący to skrypt wykonuje następujące czynności:

• wyświetla informację o wszystkich podkatalogach bieżącego katalogu, mających uprawnienia rwx dla właściciela, podlicza ile plików było modyfikowanych w tym katalogu nie wcześniej niż 1 dzień wstecz (nie więcej niż 1 dzień temu);

jeżeli nie to:

• wyszukuje wszystkie pliki w tym katalogu (wyświetlić ich nazwy), które zawierają wiersze zaczynające się od „echo”.

War.5 Przy uruchomieniu skryptu podajemy 2 parametry – słowa – jeżeli ich ilość będzie mniejsza lub większa od 2 – komunikat o błędzie i zakończenie skryptu z kodem 11. Następnie skrypt sprawdza je:

• jeżeli pierwsze słowo jest równe nazwie katalogu domowego użytkownika, a drugie równe „pies” to powinno być uruchomione polecenie „ls /”

• jeżeli pierwsze lub drugie słowo = „zapas” – komunikat „Opcja 2”.

War.6 Przy uruchomieniu skryptu podajemy 3 parametry – liczby – jeżeli ich ilość będzie mniejsza lub większa od 3 – komunikat o błędzie. Następnie skrypt wyświetla wprowadzone parametry, potem sprawdza je:

• jeżeli pierwsza liczba jest równa 11, a druga równa lub mniejsza od 3 to powinno być uruchomione polecenie „find . -type f | wc -l”

• jeżeli pierwsza lub druga liczba jest równa ilości plików zwykłych w podkatalogu, którego nazwa ma być wczytana – komunikat „Opcja 2”.

**4)** Skrypty z funkcjami.

Skrypt przy uruchomieniu pobiera 3 parametry. Następnie – w zależności od wariantu - wybór wariantu: (((<dzień urodzin> +3)\*4) mod **5** )+1 – wywołuje funkcję, która wykonuje stosowne działania:

- war.1 – funkcja pobiera jako parametry słowo „kod”, liczbę 15, drugi parametr skryptu, następnie do pliku o nazwie jak pierwszy parametr funkcji zapisuje swoje parametry: drugi i trzeci, po jej zakończeniu skrypt wyświetla zawartość tego pliku;

- war.2 – funkcja pobiera jako parametry liczby 17 oraz 31, pierwszy parametr skryptu, wyświetla swoje pobrane parametry, następnie oblicza sumę pobranych swoich pierwszego i drugiego parametru, sprawdza – czy istnieje plik w bieżącym katalogu o nazwie jak jej trzeci parametr;

- war.3 – funkcja pobiera jako parametry drugi parametr skryptu, liczbę 41, następnie wyświetla komunikat - „działa funkcja w skrypcie:” - oraz nazwę skryptu, potem wyświetla „pobrane parametry” oraz ich wartości;

- war.4 – funkcja pobiera jako parametry w takiej kolejności: drugi parametr skryptu, liczbę 7, pierwszy parametr skryptu, następnie wyświetla pobrane parametry oraz sprawdza czy pierwszy pobrany parametr nie jest równy nazwie skryptu – stosowny komunikat;

- war.5 – funkcja pobiera jako parametry: drugi parametr skryptu, słowo „kod”, liczby: 37 i 15, tworzy zmienną lokalną c21 – wpisując do niej wartość 11, następnie oblicza sumę pobranych liczb – wynik zapisuje do utworzonej w funkcji zmiennej M1, skrypt wyświetla wartości zmiennych M1 i c21 (w celu sprawdzenia – czy rzeczywiście c21 jest zmienną lokalną).